AD-A051 971 VIRGINIA POLYTECHNIC INST AND STATE UNIV BLACKSBURG --ETC F/G 20/4
THREE-DIMENSIONAL INCOMPRESSIBLE BOUNDARY LAYERS ON BLUNT BODIE--ETC(U)
MAY 77 D L DWOYER, C H LEWIS, P R GOGINENI

UNCLASSIFIED VPI/SU-AERO-063-PT-2 NL

1 OF 3
ADA
051971

VPI & SU AERO-063

Three-Dimensional Incompressible Boundary Layers On Blunt Bodies Including Effects Of Turbulence, Surface Curvature And Heat And Mass Transfer

Part II: Computer Code User's Manual

by

D. L. Dwoyer, Clark H. Lewis and P. R. Gogineni

This research was sponsored by the Applied Physics Laboratory of the Johns Hopkins University under Subcontract Number 600325.

Aerospace and Ocean Engineering Department
Virginia Polytechnic Institute and State University
Blacksburg, Virginia 24061

May 1977

Three-Dimensional Incompressible Boundary Layers On Blunt Bodies Including Effects Of Turbulence, Surface Curvature And Heat And Mass Transfer

Part II: Computer Code User's Manual

by

D. L. Dwoyer, Clark H. Lewis and P. R. Gogineni

This research was sponsored by the Applied Physics Laboratory of the Johns Hopkins University under Subcontract Number 600325.

Aerospace and Ocean Engineering Department
Virginia Polytechnic Institute and State University
Blacksburg, Virginia 24061

May 1977

ABSTRACT

The computer code that was developed as a result of the analysis of Part I of this report, VPI-AERO-063, is herein described and documented. Included are programs input, expected output, description of mode of operation and other information useful in running the program.

TABLE OF CONTENTS

Page

ABSTRACT ii

TABLE OF CONTENTS iii

LIST OF FIGURES iv

NOMENCLATURE v

INTRODUCTION 1

I. Program TAPGEN 2

1. Body Geometry and Inviscid Velocities 2

2. Description of Input Data 4

3. Description of Program Output 22

II. Program ICBL3D 32

1. Program Input 33

2. Description of Input Data 34

3. Description of Output Data 41

4. Sample Input/Output 45

APPENDIX I: Job Control Language for Program TAPGEN 52

APPENDIX II: TAPGEN Sample Input 55

APPENDIX III: TAPGEN Sample Output 58

APPENDIX IV: Program TAPGEN Data FORMAT Sheets . . . 62

APPENDIX V: TAPGEN Program Listings 71
1. Program BLOT Listing 72
2. Program DERVAT Listing 108
3. Program INVTAP Listing 113

APPENDIX VI: Job Control Language for Program ICBL3D 119

APPENDIX VII: ICBL3D Sample Input 121

APPENDIX VIII: ICBL3D Sample Output 123

APPENDIX IX: Program ICBL3D Data FORMAT Sheets . . . 138

APPENDIX X: Program ICBL3D Listing 152

LIST OF FIGURES

Figure		Page
1	Body Coordinate Systems for Inviscid and Boundary-Layer Calculations	46
2a	Block Diagram for ICBL3D Axisymmetric Bodies	47
2b	Block Diagram for ICBL3D Nonaxisymmetric Bodies	48
3	Flow Chart of Program ICBL3D	49, 50

NOMENCLATURE

h_ξ	metric coefficient of the ξ coordinate
h_ω	metric coefficient of the ω coordinate
K_ξ	$(h_\xi h_\omega)^{-1} \partial h_\xi / \partial \omega$, curvature of the ξ coordinate line
K_ω	$(h_\xi h_\omega)^{-1} \partial h_\omega / \partial \xi$, curvature of the ω coordinate line
r	local body radius in stagnation point polar coordinates
s	surface distance along ξ coordinate
U_ξ	inviscid velocity component in ξ direction
U_ω	inviscid velocity component in ω direction
U_e	inviscid velocity component in ξ direction at the body surface
w	$w = W_e$ at stagnation point and on symmetry planes, $W = U_e$ elsewhere
W_e	inviscid velocity component in ω direction at the body surface
x	distance along stagnation point axis
α_1	W/U_e
α_2	W_e/U_e
α_3	W_e/W
β_1	$(h_\xi U_e)^{-1} \ \xi \ \partial U_e / \partial \xi$
β_2	$(h_\omega U_e)^{-1} \ \xi \ \partial U_e / \partial \omega$
β_3	$(h_\omega W_e)^{-1} \ \xi \ \partial W_e / \partial \omega$
β_4	$(h_\xi W_e)^{-1} \ \xi \ \partial W_e / \partial \xi$
ξ	orthogonal surface coordinate in longitudinal direction
σ	source density
ϕ	meridional angle in stagnation point coordinate system
ω	orthogonal surface coordinate in transverse direction

INTRODUCTION

A computer program has been developed for solving the three-dimensional boundary-layer flow on blunt bodies using a finite difference procedure. The program is capable of calculating laminar or turbulent flows and accounts for the effects of surface curvature. The equations solved, finite difference procedure used and results of test calculations with the code are presented in Ref. 1.

The boundary-layer program is not a stand alone program, but requires the generation of a data tape which describes the body geometry to be solved and the surface inviscid pressure distribution over the body. This report is divided into two sections, the first describes program TAPGEN which is capable of generating the required data tape for the boundary-layer code, and program ICBL3D which actually solves the boundary-layer equations for the body of interest.

All of the programs have been coded in FORTRAN IV and written for the IBM 370/158 computer with the JES2 operating system.

SECTION I

PROGRAM TAPGEN

In this section a brief description of the computer programs used to generate the inviscid data is presented to enable the user to execute the programs easily. The input data preparation and the outputs available have been described in detail. Finally the JCL required to execute the programs on IBM/370 Model 158 with JES2 operating system is listed.

The necessary computer programs are HESS, AOAT, BLOT, DERVAT and INVTAP. Program HESS generates the body geometry and inviscid velocity distribution for two onset flows: one in x-direction and the other in y-direction. These two flows are combined by program AOAT to give the inviscid velocity distribution at the given angle of attack. Programs HESS and AOAT may be replaced by program HESS3D for calculating the inviscid flow over bodies with non-axisymmetric shapes. Program BLOT reads this velocity distribution and computes the position of surface coordinates, metric coefficients and the inviscid velocities in surface coordinates. All this information is read by program DERVAT which computes the required derivatives. Program INVTAP fits appropriate Fourier series in ϕ; computes the Fourier coefficients and writes them on a tape as required by the boundary layer programs. These programs taken together represent program TAPGEN.

1. Body Geometry and Inviscid Velocities

The boundary-layer equations have been written in an orthogonal curvilinear surface coordinate system with its origin fixed at the stagnation point. The boundary-layer computations need the body geometry, metric coefficients and the inviscid velocity distribution. These are computed separately and a brief description of these computations follows.

The surface of the body is defined in the cartesian coordinate system shown in Fig. 1 . These coordinates form the input to the computer program which computes the inviscid velocity distribution at the given angle of attack. This computer program is capable of handling axisymmetric bodies at angle of attack and three-dimensional bodies at zero-angle of attack.

The inviscid flow velocities obtained above have been transformed into a orthogonal curvilinear surface coordinate system in which boundary-layer computations were made. The position of the surface coordinate system and the metric coefficients have been determined as required by the boundary-layer program. The mathematical details of these computations can be found in Ref. 2 .

Since the position (x, r, ϕ) of the surface coordinates and the position of the inviscid velocity would not be the same in general, the inviscid velocities at the surface coordinates have been obtained by double interpolation technique in (x, ϕ). Then the results are written on a scratch tape unit for every ξ=constant. The quantities on tape are ϕ, h_ξ, h_ω, r, s, U_ξ, U_ω and x. Since the boundary-layer program needs the first derivatives with respect to ξ, of h_ξ, h_ω, U_ξ, U_ω, x, r and the second derivative of r, they have been computed using the five-point Lagrangian formula and written on the same tape. These data are fitted with appropriate Fourier series in ϕ and the Fourier coefficients have been written on another tape. It is from this tape that the boundary-layer program reads and obtains the necessary information about the geometry and inviscid velocity distribution of the body.

Immediately following is a description of the input to the inviscid program package along with sample input/output and a description of the Job Control Language required.

Basically, the calculation of the inviscid data tape is broken up into three steps as illustrated in the flow chart in Fig. 2 . First the inviscid velocity distribution is generated with either of the potential flow programs of Refs. 3 or 4 . Next the surface coordinate system is generated by program BLOT. The inviscid velocity field is then interpolated onto the surface coordinate system, Fourier fitted and written on the boundary-layer code data tape in programs DERVAT and INVTAP. Program AOAT is used only with the axisymmetric HESS code and serves to combine the zero and ninety degree angle of attack solutions into the solution at the desired angle of attack.

2. Description of Input Data

Program HESS (Axisymmetric)

A. Card Column Parameter Locations

Card 1 - Header Card.

cc1-60	Header	Any alphanumeric run description.
cc63-68	Case	Case number.

<u>Card 2</u> - Control Flag Card

cc1	NB	Number of bodies ($1\leq NB\leq 9$).
cc2	NNU	Number of non-uniform flows ($0\leq NNU\leq 5$).
cc3	IAXI	Axisymmetric flow flag.
cc4	ICRØSS	Crossflow flag.
cc5	IØFF	Off-body point input flag.
cc6	IØNLY	Basic-data-only flag.
cc7	IELPSE	Ellipse generator flag (see also Card 5).
cc8-10	(blank)	
cc11	IPRTRB	Perturbation velocities only.
cc12	IPØTNL*	Solve potential matrix.
cc14	IPTANV	Prescribed tangential velocity (for the last IPTANV bodies).
cc15	IVØRT	Strip-ring vorticity flag.
cc16	IØMITA	Omit axisymmetric uniform flow solution.
cc17	IOMITC	Omit crossflow uniform flow solution.
cc18	ISURFV	Surface vorticity (instead of sources) for the final ISURFV bodies.
cc19	IPRSCV	Prescribed values of the surface vortex strengths for the final ISURFV bodies will be input.
cc20	IALLV	All bodies are surface vorticity bodies.
cc21	IEXCRS*	Extra crossflow.
cc22	IGENBC	Generated boundary conditions.
cc23	IRNGW	Ring wing option.
cc28	IPNCH	Punched output.
cc29,30	IUNIT	Unit number for input coordinates (default = 05).

*Available if and only if NØNEWF = 1, ISIGF = 1 and IGEØMF = 1.

cc31	IVIJ	Matrix print flag.
cc32	ICØEF	Matrix-assembly coefficient print flag.
cc33	IPRINT	Very detailed matrix construction print flag.
cc34	IRAKF	Automatic rake generation flag (see also Cards 8 & 9)

Card 3 - Chord/Mach number card.

cc1-10	CHØRD	Reference chord length (default = 1.0).
cc11-20	XMACH	Mach number for Goethert correction (0.0 implies incompressible).
cc21-80	(blank)	

Card 4 - Body Control Card 1 of 2.

cc1	IGEØMF	0 = curved elements; 1 = flat elements.
cc2	ISIGF	0 = parabolic σ; 1 = linear σ; 2 = constant σ (on each element).
cc3	ICURVN	0 = internally calculated element curvatures; 1 = input curvature (see card 7).
cc4	NØNEWF	0 = use the newest formulas; 1 = use the old formulas (implies flat elements and constant σ).
cc5	IFØRMT	Input format flag (see Card Set 6).
cc6-10	NN	Number of defining endpoints for this body.
cc11-20	XMULT	x-multiplier value (default = 1.0).
cc21-30	YMULT	y-multiplier value (default = 1.0).
cc31-40	THETA	Coordinate rotation value (degrees, measured about -Z-axis.
cc41-50	ADDX	x-increment (to be applied to all the input coordinates for this body).
cc51-60	ADDY	y-increment (to be applied to all the input coordinates for this body).

Card 5 - Body Control Card 2 of 2

cc1-10	IBDN	"Body" number (sequential for bodies, zero for off-body points)	
cc11-20	ISUBKS	Subcase flag	
cc21-30	NLF	Non-lifting flag (for combination cases, only)	
cc31-40	A	Semi-major axis for ellipse cases	If IELPSE ≠ 0
cc41-50	B	Semi-minor axis for ellipse cases	

Card (Set) 6 - Body Definition Cards

IF IFØRMT=0: X-coordinates (6F10.5), then Y-coordinates (6F10.5)

IF IFØRMT=1: X, Y coordinates (2F10.5) (i.e., one "point-set" per card).

IF IFØRMT=2: X, Y coordinates (F10.5,10X,F10.5) (i.e., one "point set" per card).

Card (Set) 7 - Input curvature values (needed only if ICURVN ≠ 0)

(6F10.5) CURV(I),I=1,NN-1 The curvature values for the NN-1 elements which constitute this body

Repeat Cards 4-7 a total of (NB+IØFF) times

Card 8 - Rake Number Card (needed only if IRAKF ≠ 0)

cc1-10	NRAKES	The number of "automatically" generated mass-flow rakes

Card 9 - Rake Definition Card (needed only if IRAKF ≠ 0)

cc1-10	X1	Coordinates of "start" of the rake
cc11-20	Y1	
cc21-30	X2	Coordinates of the "end" of the rake
cc31-40	Y2	
cc41-45	N	Number of intervals to be used in the rake (note, 4≤N≤200, and N must be an even integer)

Repeat Card 9 a total of NRAKES times.

Card 10 - Non-Uniform Flow Control Card (needed only if NNU $\neq$ 0)

cc1-10 NUN Flow identification number

cc11-20 MSF 0 = axisymmetric onset flow; 1 = crossflow onset flow, 2 = both 0 and 1

cc21-30 TYPE +1.0 = velocity will be input in x,y component form; 0 = velocities will be input in normal, tangential form; -1 = automatic generation of flow due to rotation about the Z-axis (for crossflow, only)

cc31-40 FG Flow generator constant

Card (Set) 11 - Non-Uniform Flow Velocities (needed only if NNU $\neq$ 0)

(6F10.5) VX(I) or VN(I), I = 1, total number of control points

(6F10.5) VY(I) or VT(I), I = 1, total number of control points

Repeat Cards 10 and 11 a total of NNU times.

Card (Set) 12 - Specified Tangential Flow Velocities needed only if IPTANV $\neq$ 0

(6F10.5) TG(I), I = 1, total number of control points on the last IPTANV bodies

B. Discussion of Parameters

In all cases, the integer values (indicated by the standard FØRTRAN IV naming convention) must be right adjusted within the specified input fields. For floating-point type input, the input decimal point will override the quoted FØRMAT specifications.

Card 1. Self-explanatory.

Card 2. Although most of these flags are unchanged from the original program, a brief description of each one will be given here for completeness.

NB The total number of separate "bodies" ($1 \leq NB \leq 8$).

NNU The total number of (user specified) input non-uniform flows ($0 \leq NNU \leq 5$).

IAXI Axisymmetric flow flag (0: no axisymmetric flow, 1: axisymmetric flow)

ICROSS Crossflow flag (0: no crossflow, 1: crossflow)

IOFF Off-body point input flag (0: no off-body points will be input, 1: off-body points will be input)

IONLY Basic-data-only flag (0: full execution, 1: basic data only)

IELPSE Ellipse generator flag (see also card 5)(0: standard body defining coordinates will be input, 1: coordinates for the ellipse whose properties are specified on card 5 are to be created automatically by the program).

IPRTRB Perturbation velocities only (0: standard total net velocities including the onset flow will be printed, 1: perturbation velocities only will be printed).

IPØTNL Solve the potential matrix (0: standard solution for the surface velocities, 1: solution for the surface velocity potential). Since the velocity potential problem for the higher order elements has not been coded in the present version, the solution for the potential is available only for the case of flat elements and, in particular, through the usage of the "old" velocity formulas. These "old" formulas have been retained as a subsection of this new, higher order program and may be "reached" by setting the following quantities (on Card 4): IGEØMF = 1, ISIGF = 2, and NØNEWF = 1; in such a case, the solution for the potential is still available.

IPTANV Prescribed tangential velocity flag (0: standard zero normal velocity solution is to be obtained, 1: a user defined set of prescribed tangential velocities will be input on the last IPTANV bodies; obviously, IPTANV $\leq$ NB).

IVØRT Strip ring vorticity onset flows for each of the bodies (0: no strip ring vorticity, 1: automatic generation of the strip ring vorticity onset flows).

IØMITA Omit the uniform axisymmetric flow solution (0: calculate the standard uniform axisymmetric flow solution if IAXI $\neq$ 0, 1: omit this uniform axisymmetric flow solution even if IAXI $\neq$ 0).

IØMITC Omit the uniform crossflow solution (0: calculate the uniform crossflow solution if ICROSS $\neq$ 0, 1: omit the uniform crossflow solution even if ICRØSS $\neq$ 0).

ISURFV Surface vorticity for axisymmetric flow (in place of surface sources) flag (0: use the standard source distribution on all elements, ≠ 0: use vorticity instead of a source distribution as the singularity on the elements of the last ISURFV bodies). Note that if ISURFV ≠ 0, then both IVØRT and IPTANV must be ≠ 0, and then in particular, IPTANV must = ISURFV). If we assume that IPTANV = ISURFV = k, and that M is the total number of elements on the last k bodies, then the surface vorticity option causes the source induced velocity formulas to be used on the first N-M elements of each matrix row, and the vortex formulas to be used on the remaining M elements of each matrix row. The solution for the unknown strengths then proceeds as it would have if only IPTANV was nonzero.

IPRSCV Prescribed vorticity flag for axisymmetric flow (0: prescribed vorticity values will not be input, 1: prescribed vorticity values (W_i) for the last IPRSCV bodies will be input; note that if IPRSCV is ≠ 0, then both IVØRT and IPTANV must ≠ 0, and then, in particular, IPTANV must = IPRSCV). If IPRSCV is ≠ 0, then the vortex strengths of the affected elements are taken to be $W_i/4\pi$.

IALLV Total vorticity flag for axisymmetric flow (0: all elements are assumed to be source type elements (unless ISURFV ≠ 0), 1: all elements are assumed to be vortex type elements). Note that if IALLV is ≠ 0, the use of IPTANV is optional, but not mandatory.

IEXCRS Extra crossflow flag (0: no "extra" crossflow, 1: generate an "extra" crossflow (i.e., having a potential which varies as the cosine of twice the circumferential angle)). Note that if IEXCRS $\neq$ 0, then NØNEWF = 1, ISIGF = 2, and IGEOMF must = 1, since the higher order formulas for this kind of velocity potential have not been included in this program.

IGENBC Generated boundary condition flag (0: do not generate any additional boundary conditions for the crossflow case; 1: generate the onset crossflow due to rotation about an axis normal to the axis of symmetry (see also the notes for Card 10)).

IRNGW Ring wing option (0: no ring wing option, 1: use the ring wing option (see MDC Report J0741/01, April 1970 for further details)).

IPNCH Punched card output flag (0: no punched card output, 1: punched card output).

Card 3. This card is usually left blank, which results in a default chord length of unity, and no Mach number corrections (i.e., incompressible results). If a non-zero value of the Mach number is input, the program uses the Goethert correction to account for compressibility.

Card 4. Usage of the IGEOMF and ISIGF flags permit the user to "turn-off" any or all of the higher order element curvature and/or varying source density terms. The default values are curved elements with parabolically varying source density. If IGEØMF = 0, the program will automatically calculate the local element curvature values by the procedure described in Section II.A, unless ICURVN is non-zero, in which case the user must supply these curvature values (see Card 7). The value for the NØNEWF flag is ordinarily left blank (or zero), even if a flat element, constant source density solution is required. However, since certain of the original Douglas-Neumann Axisymmetric Potential Flow Program capabilities have not been made available in the higher order program (e.g., calculation of the potential (hence the added mass, etc.) the original formulas have been preserved within this version (see description of the IPØTNL and IEXCRS flags). The original capacilities can be obtained by setting IGEØMF = 1, ISIGF = 2, and NØNEWF = 1 for each of the input bodies, in which case the input instructions of Reference 5 apply.

The other parameters on Card 4 are self-explanatory, with the understanding that the order of coordinate transformations are as listed on the input.

Card 5. The "bodies" are normally loaded prior to any off-body points (although the latter require the usage of body control Cards 1 and 2, also). For this reason, the value of IBDN should be sequentially increasing beginning with unity. A non-zero value for ISUBKS means that the body definition cards for this body (Card Set 6), (say this is the ith body input under this header card) will not be included, but that the program is to use the ith set of points that were input under the previous header card. Obviously, this capability is useful only with "stacked" input cases.

Card 6. The three input formats that are available are as shown. The value of IFØRMT determines which format is used. The default format is the "old" format: x-coordinates, followed by the y-coordinates.

Card 7. These values are to be input only if ICURVN $\neq$ 0.

Card 8. This card is needed only if IRAKF $\neq$ 0. Note that $1 \leq$ NRAKES ≤ 20.

Card 9. These are the rake definition cards, which define the "start" and "end" of each mass flow rake. The sign convention that is employed is as follows: a positive mass flux means that the flow is from left to right to an observer traversing the rake from point (X1, Y1) to point (X2, Y2). For example, for an inlet without a bullet, point 1 is typically at y = 0.0, and point 2 would have a y-coordinate located on the inlet wall. Note that the x-coordinates need not be the same, i.e., a tilted rake may be used, if desired. Note also that one rake may "overlap" another if so desired, since they are each treated independently. It should be pointed out that a Simpson Rule integration is performed over the "N" intervals (and therefore N must be an even integer), with the program automatically generating N-1 "intermediate" points as "off-body points" for each input rake (program limit is 500 total off-body points). The velocity values at the first and last rake points are obtained by linear extrapolation of the two nearest values. In this way, the difficulty associated with calculating induced velocities at off-body points which lie very near to, or on, the surface is avoided. Typically, values of N between 10 and 20 appear to be satisfactory for most cases.

Card 10. Self-explanatory, except that FG, if entered is used in the following way to generate the rotation onset flow:

$$V_{x_i} = y_i$$

$$V_{y_i} = FG-x_i$$

$$V_N = V_{x_i} \sin \alpha_i - V_{y_i} \cos \alpha_i$$

$$V_T = V_{x_i} \cos \alpha_i + V_{y_i} \sin \alpha_i .$$

Card 11. Self-explanatory.

Program AOAT

Card 1. Format (3I3,F10.6)

cc1-3 IPRT = 1, writes the output of program HESS on Unit 6; = 0, does not write

cc4-6 N, Number of elements used in the program HESS

cc7-9 KPL, Number of meriodonial planes required

cc10-19 ALPD, Angle of attack in degrees

Program HESS (3-D)

Card 1.

cc	CONTENTS	NOTES
1→60	Title	Header information
61	IFLAG	= "0" for input body coordinates (this is the usual case) = "1" for generated ellipsoids = "2" for generated bodies with elliptical cross section
62	LIST	= "0" for full execution = "1" for basic data listing only
63	KX	= "F" or blank for no generated uniform x-flow = "T" for generated uniform x-flow (calculate x-flow matrices)
64	KY	= "F" or blank for no generated uniform y-flow = "T" for generated uniform y-flow (calculate y-flow matrices)

cc	CONTENTS	NOTES
65	KZ	= "F" or blank for no generated uniform z-flow = "T" for generated uniform z-flow (calculate z-flow matrices)
66	ISIG	= "0" for no input guesses for source densities = "1" for input guesses for source densities (rarely used)
67	IPRS	= "0" for no mid-iteration print = "1" for print of solution iterations
68	MPR	= "0" for no intermediate matrix print = "1" for V_{ij} print = "2" for A_{ij} print = "3" for V_{ij} and A_{ij} print (Never use for large case, or output is excessive)
69	METHØD	= "0" for automatic solution selection = "1" for direct matrix solution = "2" for modified Seidel iterative solution
70→71	NNØN	Number of input onset flows
72	NSYM	Number of symmetry planes
73	NØFF	= "0" for no off-body points = "1" for off-body points
74	KMACH	= "0" for no Mach number correction = "1" for input Mach number

cc	CONTENTS	NOTES
75-76	---	
77-80	KASE	Alpha-numeric case identification

Mach Number is input on a separate card only if KMACH = 1.

INPUT GENERATOR DATA

Card 2.

cc	CONTENTS	NOTES
1 →5	NLM1	Number of "latitudinal" element divisions
6→10	MMIN	Number of "longitudinal" element divisions
11→20	B	Y semi-axis of elliptical cross section
21→30	C	Z semi-axis of elliptical cross section

(If ellipsoid is generated, X semi-axis is unity)

NOTE: This card is input only when IFLAG ≠ 0

X-Z INPUT

This input consists of the coordinate pairs (X_1, Z_1, X_2, Z_2, . . . X_{NLM1+1}, Z_{NLM1+1}) that define the profile in the x-z plane of the body of elliptical cross-section to be generated by the program. These coordinates are read with an 8F10.0 format.

ON-BODY COORDINATES

General (non-elliptical) bodies are input by giving the coordinates of a large number of points on the body. The three dimensional coordinates defining the body surface(s) are input 2 points-to-a-card. Each point has associated with it a quantity called its "status" that has the following meaning:

VALUE	MEANING
"0" or blank	This point is on the same n-line as the last point
1	This point starts a new n-line
2	This point starts a new section
3	This is the last point of the input

NOTE: These cards are only input if IFLAG=0

OFF-BODY POINT COORDINATES

The three-dimensional coordinates defining the off-body points are input with the identical format of the on-body coordinates. The only status flag used, however, is that of "3" for the last off-body point.

NOTE: These cards are input only if NØFF=1.

FLOW FLAGS

Each input onset flow is preceded by a card that describes the flow. All x-flows <u>must</u> precede all y-flows which precede all z-flows. (An "x-flow" is any flow that is solved using the matrix appropriate for a uniform onset flow parallel to the x-axis.)

cc	MEANING
1	= "1" if this is an x-flow = "2" if this is a y-flow = "3" if this is a z-flow
2	= "0" if this is a non-uniform onset flow, i.e., values of the onset flow velocities will be specified for each element = "1" if this is a uniform onset flow, the value of which is only specified once
3	= "0" if the onset flow velocity is expressed in component form = "1" if the onset flow is expressed in normal velocity values

NOTES: The onset flow velocities and sigma guesses are read with a "6F10.0" format.

This input is only required if NNØN = 0.

GENERAL PROCEDURES

All coordinates are input with the following format

2 (3F10.0, I1)

Except where noted elsewhere, zeroes or blanks may be interchanged on input.

Program BLOT

Card Name	Fortran			
Col. 1-5	Name	Format	Column	Description
TITLE	TITLE	9A8	6-77	Problem Identification - this will appear as a heading in the output
LIMTS	JMAX	I5	6-10	Number of points across boundary layer
	KMAX	I5	11-15	Number of points around body (in phi-direction)
	LITER	I5	16-20	Number of iterations for initial profile
	IPRT	I5	21-25	Every how many steps printing will occur along the body
	KPRT	I5	26-30	Every how many steps printing will occur around the body
	PSIMAX	E10.0	31-40	Distance along body at which program will terminate
	UINF	E10.0	41-50	Freestream velocity - V_∞
DW	DW(K)	E10.0	6-15	$\Delta\omega$'s around the body, K = 1, KMAX - 1.
			16-25	
			.	
			.	
			.	
			etc.	
DN	DN(J)	E10.0	6-15	$\Delta\eta$'s across the boundary layer. J = 1, JMAX - 1.
			16-25	
SIZE	A	E10.0	6-15	a - length of axis of triaxial ellipsoid ($A>10^6$ gives paraboloids)

Card Name Col. 1-5	Fortran Name	Format	Column	Description
	RB	E10.0	16-25	R_b - radius of curvature of nose of body in the plane y'= 0
	RC	E10.0	26-35	R_c - radius of curvature of nose of body in the plane z' = 0
	EPS	E10.0	36-45	Convergence criterion used in calculating the body coordinates
	ALP	E10.0	46-55	α - angle of attack (deg)
	CAPTH	E10.0	56-65	Θ - indicates which finite-difference scheme is being used. $= \begin{cases} 1 = \text{Kruase scheme} \\ 2 = \text{Crank-Nicolson scheme} \end{cases}$
	CRI	E10.0	66-75	$= \begin{cases} 1 \text{ Implicit scheme} \\ 1/2 \text{ Crank-Nicolson scheme} \end{cases}$

Programs DERVAT and INVTAP

These programs obtain the necessary data sets internally and they do not need any external input.

3. Description of Program Output

Program HESS(Axisymmetric)

Unit 6

The main output of the program, which consists of calculated surface velocities and/or pressures has not been changed. Its format is that of Ref.2. The surface coordinates with which the calculated velocities are associated are the transformed coordinates, i.e., the original input coordinates altered by any specified translations, multiplications, or rotations. Similarly, the format of the calculated velocities at the off-body points is similar to that of Ref.5.The only change is that output at rake points (q.v.) is included.

The initial output, which details the surface geometry, has been changed considerably. The header identifies what kind of solution has been computed: (1) curved or flat elements; (2) constant, linear, or parabolic source density element curvatures. The header also specifies whether "new" or "old" velocity formulas have been used, the latter of which apply only to the flat-element constant-source case. In the body of the output, the first two columns are the untransformed (input) coordinates, and the third and fourth columns are the transformed coordinates, which are the endpoints of the surface elements. The fifth and sixth columns are the coordinates of the control points of the elements. Element arc lengths are given in column 7 and a running total of arc length is displayed in column 8. Column 9 lists the differences between average slopes of successive elements, i.e., the differences in the slopes of the two straight lines, through the respective element endpoints. The actual slope discontinuity at an endpoint between two parabolic elements is normally much smaller than the difference between the average slopes of the two elements. Moreover, this discontinuity approaches zero if the body contour approaches a parabola. The final (tenth) column lists elements curvatures.

A new output is entitled Automatic Rake Calculation. For each rake the two input points that bound the rake and the input identification are output together with three calculated quantities. The first is the surface area of the cone frustum defined by the two input rake points. The third is the total flux of fluid that crosses this area per unit time. The second quantity, average velocity, is the ratio of the third and the first quantities. There is a rake output for all axisymmetric flows, both uniform and non-uniform, but there is no rake output for cross-flow, because its circumferential variation guarantees zero flux.

With regard to surface vorticity solutions, there are only two possibilities. Either there are no vorticity solutions or there is one solution for each body, which corresponds to a unit vorticity strength on that body and zero vorticity strength on all other bodies. The order of these solutions is the same as the order in which the bodies are input. Thus, in particular, an inlet with centerbody has two vorticity solutions - one with vorticity on the inlet and one with vorticity on the centerbody. The second of these is not meaningful and should be discarded. (It may, of course, occur first on the output.)

Units 1- 4, 7-15 These are scratch disks, which are used to write and read internally.

Unit 16 The coordinates x, y and the velocities T_1, T_1, T_2 are written in order one after the other for all axial stations.

Program AOAT

Unit 6 For each axial station (x), the pressure coefficient (CP), the velocities UX, UY, UZ are written. These velocities are in body axes. Finally, the stagnation point location XO is printed.

Unit 10 For each axial station, two records are written unformated. The first record contains X and KPL. The second record contains the meridonial plane angle, UX, UY, UZ and radius R for all planes.

Unit 22 This is not used.

Unit 30 Formated image of Unit 10.

Program BLOT

Unit 6 Initially all input data is printed out. This includes several quantities which are computed strictly from the input. Next, initial profile calculations, including all iterations are printed. Finally, the solution at each desired location along and around the body is displayed. At each point where printing occurs, the following quantities appear:

$$\left.\begin{array}{c}\text{X}\\ \text{Y}\\ \text{Z}\end{array}\right\}\ \text{x,y,z coordinates}$$

R		r, radial distance to body surface
S		s, distance along surface in ξ-direction
ALPHA	$\bar{\alpha} = w_e/u_e$	
ALBET2	$\bar{\alpha}_2 = w_e/u_e$	
ALBET3	$\bar{\alpha}_3 = w_e/W$	
XIKA		parameter $\xi \ K_\xi/\bar{\alpha}$
UE	u_e	inviscid edge velocities in ξ and ω directions
WE	w_e	
BETA1	β_1	inviscid velocity gradient parameters
BETA2	β_2	
BETA3	β_3	
BETA4	β_4	
DFDN		Not calculated
DGDN		
HPSI	h_ξ	Metric coefficients
HOMEGA	h_ω	
KAPPAS	K_ξ	Curvatures of the coordinate lines
KAPPAW	K_ω	
TOTAL SHEAR STRESS		Not calculated
TOTAL SKIN FRICTION		
ANGLE		

Unit 8	Formated image of Unit 25
Unit 11	The diagnostic printout from the double interpolation subroutine SLOALL. Useful for debugging only.

Unit 25 — For each (PSI) the longitudinal surface coordinate, the metric coefficients (HPSI, HW), R, S, UPSI, UW are written for each meridonial plane, unformated.

Program DERVAT

Unit 6 — The angle of attack in radians and the number of meridonial planes are written. Next the image of Unit 25 of program BLOT has been written.

Unit 26 — The angle of attack in radians and KPL are written. Next, the station No. I, XS, HPSI, HW, R, S, UPSI, UW, X, and the derivatives of these D(UPSI), D(UW), D(R), the second derivative D2(R); D(HPSI), D(HW), D(X), are written in order for each meridonial plane. These are written unformated.

Unit 29 — Formated image of part of Unit 26. The data written is I, XS, S, X, D(X), UPSI, D(UPSI), UW, and D(UW) for each plane.

Unit 30 — Formated image of part of Unit 26. First the angle of attack in radians and KPL are written. Next I, XS, HPSI, D(HPSI), HW, D(HW), R, D(R) and D2(R) are written for each plane.

Program INVTAP

Unit 6 Errors and messages are written on this unit.

Unit 10 For each station the Fourier coefficients are written unformated. This is the data set required by ICBL3D.

Unit 30 Formated image of Unit 10. Also at each station the data written on Unit 26 of program DERVAT is rewritten.

JCL

The programs described above require large memory, which made it necessary to execute them one after the other in a multi-step job. Also the programs were compiled, linkedited and stored as members of a partition data set, named A30303. MARK1. SUBLIB. This has the advantages of avoiding handling large Fortran source decks and compiling everytime the job is executed. The input cards for each program have to be inserted after the //FT05F001 DD* card, except for the program BLOT for which they have to be after the //FT04F001 DD* card. Finally, the required inviscid data has been written on an on line disk (CHEKOV) as a partition data set named A505F3.INVTAP. The relevant JCL has been listed below.

```
1        Job Card
2        //JOBLIB DD DSN=A30303,MARK1,SUBLIB,DISP=(SHR,KEEP),
         //VOL=SER=USERPK,UNIT=SYSDA
3        //STEP1 EXEC PGM=HESS,TIME=1,REGION=520K
4        //FT06F001 DD DUMMY
```

```
5    //FT01F001 DD UNIT=SYSDA,DISP=(NEW,DELETE),SPACE
     =(TRK,(50,1))

6    //FT02F001 DD UNIT=SYSDA,DISP=(NEW,DELETE),SPACE
     =(TRK,(50,1))

7    //FT03F001 DD UNIT=SYSDA,DISP=(NEW,DELETE),SPACE
     =(TRK,(50,1))

8    //FT04F001 DD UNIT=SYSDA,DISP=(NEW,DELETE),SPACE
     =(TRK,(50,1))

9    //FT07F001 DD DUMMY

10   //FT08F001 DD UNIT=SYSDA,DISP=(NEW,DELETE),SPACE
     =(TRK,(50,1))

11   //FT09F001 DD UNIT=SYSDA,DISP=(NEW,DELETE),SPACE
     =(TRK,(50,1))

12   //FT10F001 DD UNIT=SYSDA,DISP=(NEW,DELETE),SPACE
     =(TRK,(50,1))

13   //FT11F001 DD UNIT=SYSDA,DISP=(NEW,DELETE),SPACE
     =(TRK,(50,1))

14   //FT12F001 DD UNIT=SYSDA,DISP=(NEW,DELETE),SPACE
     =(TRK,(50,1))

15   //FT13F001 DD UNIT=SYSDA,DISP=(NEW,DELETE),SPACE
     =(TRK,(50,1))

16   //FT15F001 DD UNIT=SYSDA,DISP=(NEW,DELETE),SPACE
     =(TRK,(50,1))

17   //FT16F001 DD UNIT=SYSDA,DISP=(NEW,PASS),SPACE
     =(TRK,(50,1)),DSN=&&T1

18   //FT05F001 DD *

19   //STEP2 EXEC PGM=AOAT,COND=(0,NE),TIME=(0.30),
     REGION=150K

20   //FT06F001 DD DUMMY

21   //FT16F001 DD UNIT=SYSDA,DISP=(OLD,DELETE),DSN=&&T1

22   //FT22F001 DD DUMMY

23   //FT30F001 DD DUMMY

24   //FT10F001 DD UNIT=SYSDA,DISP=(NEW,PASS),SPACE=
     (TRK,(50,1)),DSN=&&T2
```

```
25        //FT05F001 DD *

26        //STEP3 EXEC PGM=BLOT,COND=(0,NE),TIME=5,REGION=350K

27        //FT06F001 DD SYSOUT=A

28        //FT08F001 DD SYSOUT=A

29        //FT11F001 DD DUMMY

30        //FT10F001 DD UNIT=SYSDA,DISP=(OLD,DELETE),DSN=&&T2

31        //FT25F001 DD UNIT=SYSDA,DISP=(NEW,PASS),SPACE=
          (TRK,(50,1)),DSN=&&T3

32        //FT04F001 DD *

33        //STEP4 EXEC PGM=DERVAT,COND=(0,NE),TIME=1,REGION=650K

34        //FT06F001 DD SYSOUT=A

35        //FT25F001 DD UNIT=SYSDA,DISP=(OLD,DELETE),DSN=&&T3

36        //FT26F001 DD UNIT=SYSDA,DISP=(NEW,PASS),SPACE=
          (TRK,(50,1)),DSN=&&T4

37        //FT29F001 DD SYSOUT=A

38        //FT30F001 DD SYSOUT=A

39        //STEP5 EXEC PGM=INVTAP,COND=(0,NE),TIME=1,REGION=650K

40        //FT06F001 DD SYSOUT=A

41        //FT25F001 DD UNIT=SYSDA,DISP=(OLD,DELETE),DSN=&&T4

42        //FT10F001 DD UNIT=SYSDA,DISP=(NEW,CATLG),SPACE=
          (TRK,(50,1)),
          // DSN=A505F3.INVTAP,VOL=SER=CHEKOV

43        //FT30F001 DD SYSOUT=A
```

JCL for use with Fortran source decks

In the JCL listed above the following changes have to be made:

1. Card 2 has to be removed.

2. In card 3 (PGM = HESS) has to be replaced by (FORTGCG)
   19 (= AOAT) has to be replaced by (FORTGCG)
   26 (= BLOT) has to be replaced by (FORTGCG)
   33 (= DERVAT) has to be replaced by (FORTGCG)
   39 (= INVTAP) has to be replaced by (FORTGCG)
3. After cards 3, 19, 26, 33, 39 insert the following card:

   //FORT.SYSIN DD* and then appropriate Fortran source deck.

Listings of Job Control Language, sample data decks, sample output and program listings for the TAPGEN package are found in Appendices I through V.

SECTION II

PROGRAM ICBL3D

In this section program ICBL3D is described. The general structure of the program is briefly described and then the input data and output of the program are described in detail. Finally, the Job Control Language required to execute the program on an IBM 370/158 computer with the JES2 operating system is listed.

A flow chart of the program appears in Fig. 3 where the function of the various subroutines are indicated. As can be seen, the code requires a small amount of card input, but its' primary mode of data input is from a data tape designated as logical unit 10. This data tape can be prepared using program TAPGEN described earlier in this report.

The program proceeds in a forward marching fashion from the body stagnation point downstream and from the windward to leeward symmetry planes. The number of solution planes around the body in the ω-direction is fixed by the user, while the step size in the ξ-direction is adjusted internally by the program as it proceeds downstream. The initial step size is set by the user and this step size is continually adjusted by the program based upon the number of iterations required to achieve a solution at a previous station. Further, at the user's option, the edge value of the transformed normal coordinate η can be automatically increased or decreased to meet a predetermined criteria for profile decay at the boundary-layer outer edge.

There is a special feature of the program for full three-dimensional solutions. If the program cannot obtain a converged solution at some point between the windward and leeward symmetry planes it will drop that point (and all subsequent points to the leeward plane) from the solution around the body at all subsequent streamwise stations. This feature allows the solution to proceed even after separation or other problems are encountered near the leeward plane of the body.

1. Program Input

As mentioned, input to the program is from two different sources: (1) cards or card image data and (2) data sets residing on tape or disk. The data sets residing on tape or disk are generated by program TAPGEN and basically consist of Fourier coefficients. The first record on the tape contains the angle of attack in radians and an integer variable giving the number of coefficients in each Fourier series. All the records on the file after the first are identical in format. They consist of the value of ξ followed by the Fourier coefficients for the series fits to r, U_e, W_e, h_ξ, h_ω, s, $\partial U_e/\partial\xi$, $\partial W_e/\partial\xi$, $\partial r/\partial\xi$, $\partial^2 r/\partial\xi^2$, $\partial h_\xi/\partial\xi$, $\partial h_\omega/\partial\xi$, x, $\partial x/\partial\xi$ in that order. The series are in terms of the angle ϕ and their range is from the windward to leeward planes of symmetry for constant ξ. All data on the tape are written with unformatted WRITE's. The data must appear on logical unit 10.

The card or card image data to the program will now be described.

2. Description of Input Data

CARD 1. LABEL (20A4)

LABEL is the title of the case and is a single subscripted array. LABEL is passed to the output routines and appears as the header to the program output.

CARD 2. IE (49X,I3)

IE is the number of points taken normal to the body. The program is dimensioned for a maximum of 101 points in the normal profile arrays. 101 is the recommended value; however, savings in execution time occur by decreasing IE. IE must be an odd number.

CARD 3. INJCT (49X,I3)

INJCT is the subscript of the XSTA array (see below) giving the surface location at which mass injection begins. A zero value of INJCT is set to NSOLVE.

CARD 4. KADETA (49X,I3)

KADETA is an indicator for the adjustment of the transformed normal coordinate. If KADETA = 0, the maximum value of η is held constant. If KADETA = 1 the maximum η is adjusted when the velocity profiles fail to decay properly at the outer edge as prescribed by the input variable ADTEST. The value of η_{max} as adjusted up or down for windward symmetry plane only runs. For full three dimensional runs η_{max} is only adjusted up, and can be adjusted at every point. A value of 1 is recommended.

CARD 5.   KEND2   (49X,I3)

KEND2 specifies the number of circumferential planes to be used in the solution, and thereby also sets the circumferential step size. The program can accept a maximum value of KEND2 = 61. A total of sixty-one planes would lead to excessive computing time, however, and a value of KEND2 = 11 is recommended.

CARD 6.   KONSET   (49X,I3)

KONSET is the subscript of the XSTA array giving the location of the onset of transition. At X = XSTA(KONSET) the variable LAMTRB is reset to 2 and transition to turbulence begins. A zero value is reset to NSOLVE.

CARD 7.   KPRT   (49X,I3)

KPRT is a print control parameter which controls the printing of profiles in the ϕ direction. The program prints every KPRT'th profile around the body at a given value of ξ starting with the $\phi = 0^o$ profile.

CARD 8.   KTRANS   (49X,I3)

KTRANS is an indicator for the transition model. If KTRANS = 0 transition to turbulence will be instantaneous, if KTRANS = 1 a smooth transition to turbulence will take place over the distance XBAR (see below).

CARD 9.   LAMTRB   (49X,I3)

LAMTRB indicates whether the flow is laminar or turbulent. LAMTRB = 1 indicates the solution <u>begins</u> with laminar flow. LAMTRB = 2 means fully turbulent flow. When KONSET $\neq$ 0, LAMTRB = 1.

CARD 10. LPRT (49X,I3)

LPRT is the print control parameter in the streamwise direction. The program prints solutions at every LPRT'th station in ξ.

CARD 11. NIT1 (49X,I3)

NIT1 is an iteration counter used to adjust the streamwise step size. If the total number of iterations required to obtain the solution at a point is less than or equal to NIT1 then the ξ step size is doubled.

CARD 12. NIT2 (49X,I3)

NIT2 is an iteration counter used to adjust the streamwise step size. If the total number of iterations required to obtain a solution at a point is greater than NIT1 and less than NIT2 the ξ step size is unchanged.

CARD 13. NIT3 (49X,I3)

NIT3 is an iteration counter affecting ξ step size and the convergence of the solution. If the total number of iterations required for a solution at a particular point is greater than NIT3 the program halves the ξ step size and cuts back the value of ξ by the new step size. A solution at the smaller value of ξ is then tried. If this procedure fails three consecutive times execution is terminated.

NOTE: The ξ step size is adjusted only at the windward symmetry plane.

CARD 14. NOINJ (49X,I3)

NOINJ is the subscript of the XSTA array giving the surface location at which injection ends. NOINJ = 0 is reset to NOINJ = NSOLVE.

CARD 15. SFC (49X,A3)

SFC is a literal variable coded as either YES or NO to indicate whether surface curvature effects are to be included in the calculations.

CARD 16. NSOLVE (49X,I3)

NSOLVE is the number of variables in the XSTA array, and is therefore the subscript of the last XSTA value which indicates the end of the body. It is also the default value for INJCT, NOINJ, and KONSET.

CARD 17. ADTEST (49X,E14.6)

ADTEST is used in conjunction with KADETA. When KADETA is 1 ADTEST provides the convergence criteria for checking the streamwise velocity profile. When $|F(IE) - F(IE-4)|$ or $|G(IE) -G(IE-4)|$ is less than ADTEST/10 the maximum value of η is decreased by 10%. When it is greater than ADTEST the maximum value of η is increased by 10%.

CARD 18. AKSTAR (49X,E14.6)

AKSTAR is a numerical constant in the Van Driest inner eddy viscosity law (k* in Ref. 1). The recommended value is 0.435.

CARD 19. ALAMDA (49X,E14.6)

ALAMDA is a numerical constant in the outer eddy viscosity law used in the program (λ in Ref. 1). The recommended value is 0.09.

CARD 20. ASTAR (49X,E14.6)

ASTAR is a numerical constant used in the damping term of Van Driest's inner eddy viscosity law in the program (A* in Ref. 1). The recommended value is 26.0.

CARD 21. CWALL (49X,E14.6)

CWALL is the injection rate for mass transfer cases where the rate is a constant. CWALL = v_w/u_∞.

CARD 22. CONV (49X,E14.6)

CONV is the solution convergence criterion. The dependent variable arrays of streamwise and cross flow velocities are all checked for convergence at all points. When the largest percentage difference between the current and previous iterations is less than or equal to CONV the solution is taken to be converged.

CARD 23. DXMAX (49X,E14.6)

DXMAX is the maximum step size, to be taken in the streamwise direction.

CARD 24. DX1 (49X,E14.6)

DX1 is the initial streamwise step size. Since this value will be adjusted internally it is not critical that the user choose an accurate value. Usually a value of 0.01 is a good initial DX.

CARD 25. EDYLAW (49X,A3)

EDYLAW specifies the inner eddy viscosity law to be used in turbulent cases. Two options are available to the user: (1) EDYLAW = VAN DRIEST and (2) EDYLAW = REICHARDT. The program picks up only the first 3 letters of each name. The user should note the comments on these two laws for mass transfer problems stated in Ref. 1. In general the REICHARDT law is recommended.

CARD 26. ETAFAC (49X,E14.6)

ETAFAC controls the normal grid spacing. A value of 1.0 gives an equally spaced grid for the transformed normal coordinate. A value greater than 1.0 gives a finer grid at the wall than at the outer edge. A value of 1.04 is recommended with IE = 101 and ETAINF = 6.0. A value of 1.09 is recommended with IE = 101 and ETAINF = 100.0.

CARD 27. ETAINF (49X,E14.6)

ETAINF is the maximum value of η. A value between 6.0 and 10.0 is recommended for laminar flow. A value between 10.0 and 100.0 is recommended for turbulent flow. This value can be adjusted internally by specifying KADETA = 1.

CARD 28. PR (49X,E14.6)

PR is the value of the laminar Prandtl number in the fluid.

CARD 29. RTW (49X,E14.6)

RTW is the ratio of the wall temperature to stagnation temperature. It is used to calculate wall temperature when wall temperature is held constant.

CARD 29. TFS (49X,E14.6)

TFS is the free-stream static temperature in degrees Rankine.

CARD 30. CP (49X,E14.6)

CP is the specific heat of the fluid in $ft^2/sec^2 {}^oR$.

CARD 31. AMUINF (49X,E14.6)

AMUINF is the coefficient of viscosity in slugs/ft-sec.

CARD 32. PSTAG (49X,E14.6)

PSTAG is the free-stream total pressure in lbf/ft^2.

CARD 33. PINF (49X,E14.6)

PINF is the free-stream static pressure in lbf/ft^2.

CARD 34. XBAR (49X,E14.6)

XBAR is the relative length of the transition regime in turbulent cases. It is the ratio of the transition end point location to the transition onset distance.

CARD 35. UFS (49X,E14.6)

UFS is the free-stream velocity in ft/sec.

CARD 36 to 36+NSOLVE XSTA(I), I = 1, NSOLVE (F12.6)

XSTA is an important input array. It is an array of surface distances solutions calculated. The program will always obtain solutions at these points regardless of internal adjustments to the streamwise step size. Both the value 0.0 and the end point of the body must be included in the array as well as any distances describing the beginning or end of injection and transition.

CARD 36+NSOLVE+1 TO LAST CARD XTW(I), TWX(I), XCI(I), CIX(I)
(4E12.6)

TWX(I) is the wall temperature in degrees Rankine at XTW(I) which is a surface distance.

CIX(I) is the injection rate v_w/u_∞ at XCI(I) which is a surface distance.

These arrays are dimensioned for a maximum of 500 values each. If none of these cards appear in the input deck the program will automatically assume constant wall temperature and injection rate values based on RTW and CWALL. This input allows the wall temperature distribution and injection rate distribution to be read in versus their own surface distance tables. If both distributions are to be read in versus the same distance table, then either one of the two distance tables may be left blank. Another important feature is the fact that the distributions need not cover the same surface distance. For instance, the wall temperature distribution might be defined over the entire body while the injection rate distribution might only be defined over a short distance.

3. Description of Output Data

The output from program ICBL3D is in the form of printed output. The printed output is presented in two forms, station data and profile data. This section will define the output by listing the variable name as it appears in the output along with the definition of the variable. The program also prints miscellaneous messages which are described at the end of this section.

Station Data

Line 1

XO distance from the stagnation point along the body axis non-dimensionalized by the reference length (one foot)

R radial distance from the stagnation point body axis to the body surface non-dimensionalized by the reference length

PHI meridional angle about the stagnation point body axis measured from the windward symmetry plane in degrees

Line 2

XI longitudinal surface coordinate ξ

DXI $\Delta\xi$, streamwise step size

CWALL local injection rate v_w/u_∞

NIT number of iterations to obtain the solution

Line 3

HX metric coefficient h_ξ evaluated at the body

HW metric coefficient h_ω evaluated at the body

Line 4

TE nondimensional edge static temperature, $T_\infty C_p/u_\infty^2 = T_e$

UE nondimensional ξ component of velocity at the boundary-layer edge $u_e/u_\infty = U_e$

VE nondimensional ω component of velocity at the boundary-layer edge $w_e/u_\infty = W_e$

Line 5

DUEDX $\partial U_e/\partial \xi$

DVEDX $\partial W_e/\partial \xi$

DUEDW $\partial U_e/\partial \omega$

DVEDW $\partial W_e/\partial \omega$

Line 6

LOCAL EDGE REYNOLDS NUMBER = $\rho_\infty u_e \xi/\mu_\infty$

Line 7

CFXINF $C_{f_{\xi_\infty}} = 2\tau_{w_\xi}/\rho_\infty u_\infty^2$, streamwise skin friction coefficient

CFXEDG $C_{f_{\xi_e}} = 2\tau_{w_\xi}/\rho_\infty u_e^2$, streamwise skin friction coefficient based on edge conditions

CFWINF $C_{f_{\omega_\infty}} = 2\tau_{w_\omega}/\rho_\infty u_\infty^2$, transverse skin friction coefficient based on free-conditions

CFWEDG $C_{f_{\omega_e}} = 2\tau_{w_\omega}/\rho_\infty u_e^2$, transverse skin friction coefficient based on edge conditions.

Line 8

QW $q_w/\rho_\infty u_\infty^3$, wall heat transfer coefficient

CHIMAX maximum vorticity Reynolds number $n^2/\nu \quad \partial u/\partial n$

Line 9

LONGITUDINAL SKIN FRICTION τ_{ω_ξ} in lbf/ft^2

DELTA*(X) δ_ξ^*, streamwise boundary-layer displacement thickness in feet.

THETA(X) θ_ξ, streamwise boundary-layer momentum thickness in feet.

Line 10

TRANSVERSE SKIN FRICTION τ_{w_ω}, in lbf/ft^2

DELTA*(PHI) δ^*_ω, transverse boundary-layer displacement thickness in feet.

THETA(PHI) θ_ω, transverse boundary-layer momentum thickness in feet.

Line 11

WALL HEAT TRANSFER RATE q_ω in Btu/ft^2/sec

DELTA(FT) δ, the boundary-layer thickness in feet.

Profile Data

One group of boundary layer profile data is printed by the program. Every other point is printed in the profile arrays. The following columns are identified:

ETA η, the transformed normal coordinate.

Y n, the physical normal distance nondimensionalized by the reference length.

F $h_\xi u/h_{\xi,0}U_e$

FN $\partial F/\partial\eta$

G $h_\omega w/h_{\omega,0}W$ where $W = W_e$ at the stagnation point and symmetry planes and $W = U_e$ elsewhere.

GN $\partial G/\partial\eta$

T T/T_e

TN $\partial(T/T_e)/\partial\eta$

V transformed normal velocity profile.

EPLUS ε^+ the eddy viscosity coefficient

Miscellaneous Messages

The program has a few internal messages which are written to indicate problems with the solution, or coordinate adjustments. A message is printed by subroutine ADDETA whenever η_∞ is adjusted up or down. The direction of adjustment is given along with X, the old η_∞ and the new η_∞.

A message is printed by subroutine CHANGX indicating the beginning of transition or mass transfer. Included in the messages are the values of X and the particular integer counter involved. A similar message is also printed by CHANGX when mass transfer ends.

Whenever the program fails to obtain a converged solution within NIT3 iterations, a message is printed by subroutine CONTRL to that effect which includes the values of the transverse and streamwise solution counters and NIT. If this should occur three consecutive times, a message will be printed indicating that execution is terminating.

If a particular boundary-layer problem drops all of its circumferential solution planes due to convergence problems, a message will be printed by CONTRL indicating that execution is terminating.

A normal termination of the program is indicated by the message "THE END" printed out after the last station results.

3. Sample Input/Output

Listings of Job Control Language, sample input decks, sample outputs and format sheets for the input data cards and the program listings are found in Appendices VI through X.

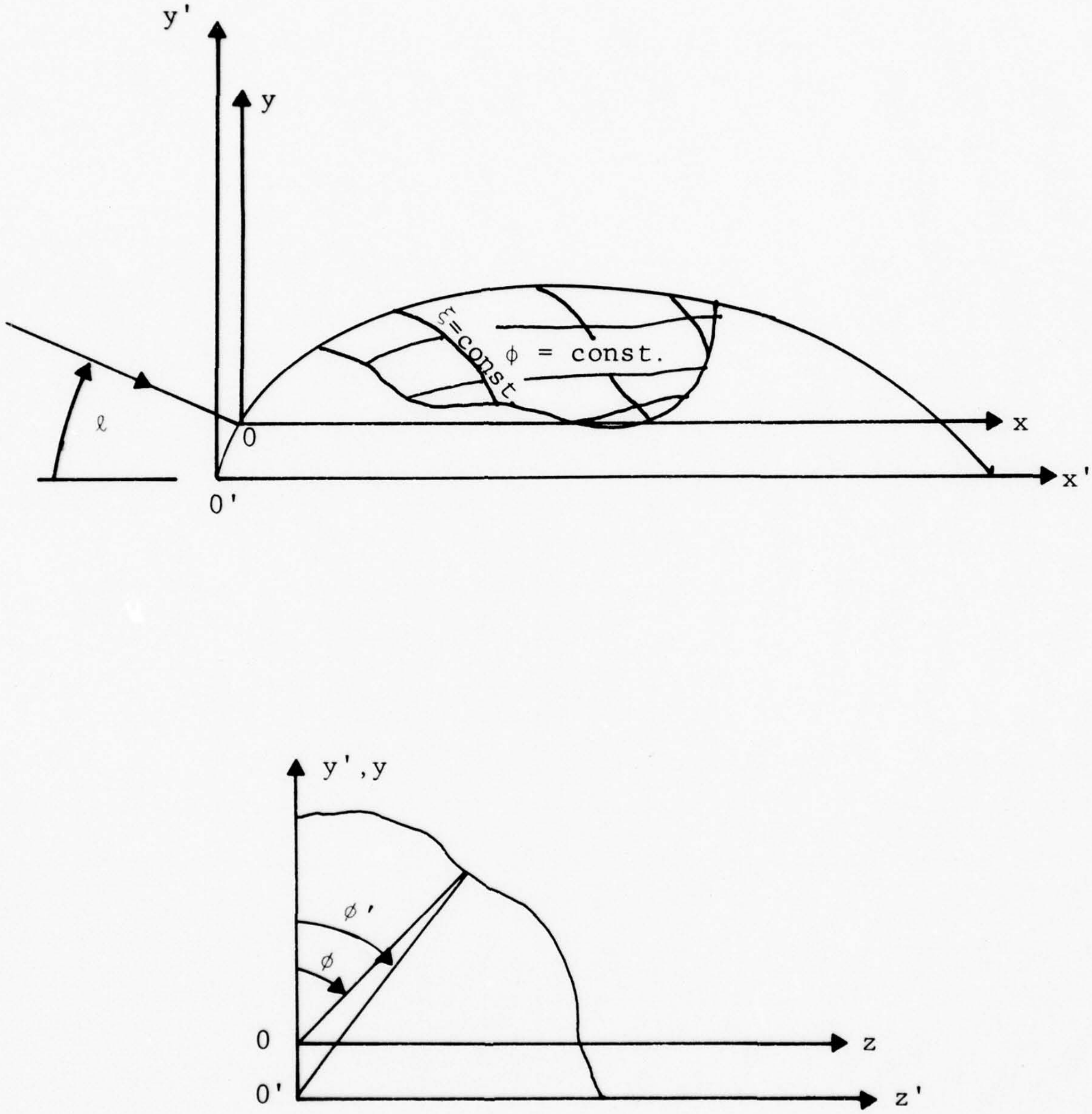


Figure 1. BODY COORDINATE SYSTEMS FOR INVISCID AND BOUNDARY-LAYER CALCULATIONS

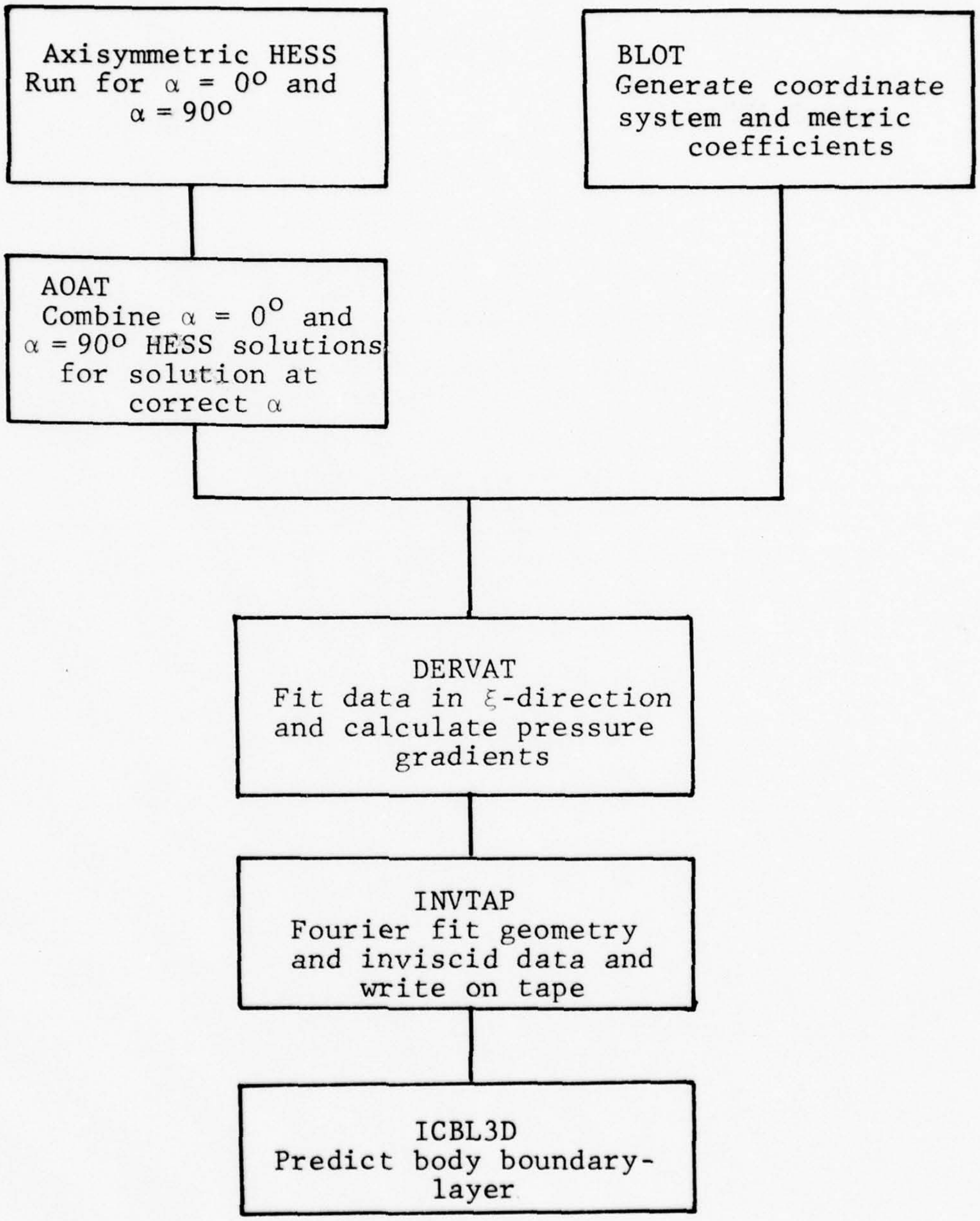

Axisymmetric HESS Run for α = 0° and α = 90°
BLOT Generate coordinate system and metric coefficients
AOAT Combine α = 0° and α = 90° HESS solutions for solution at correct α
DERVAT Fit data in ξ-direction and calculate pressure gradients
INVTAP Fourier fit geometry and inviscid data and write on tape
ICBL3D Predict body boundary-layer

Figure 2a. BLOCK DIAGRAM FOR ICBL3D AXISYMMETRIC BODIES

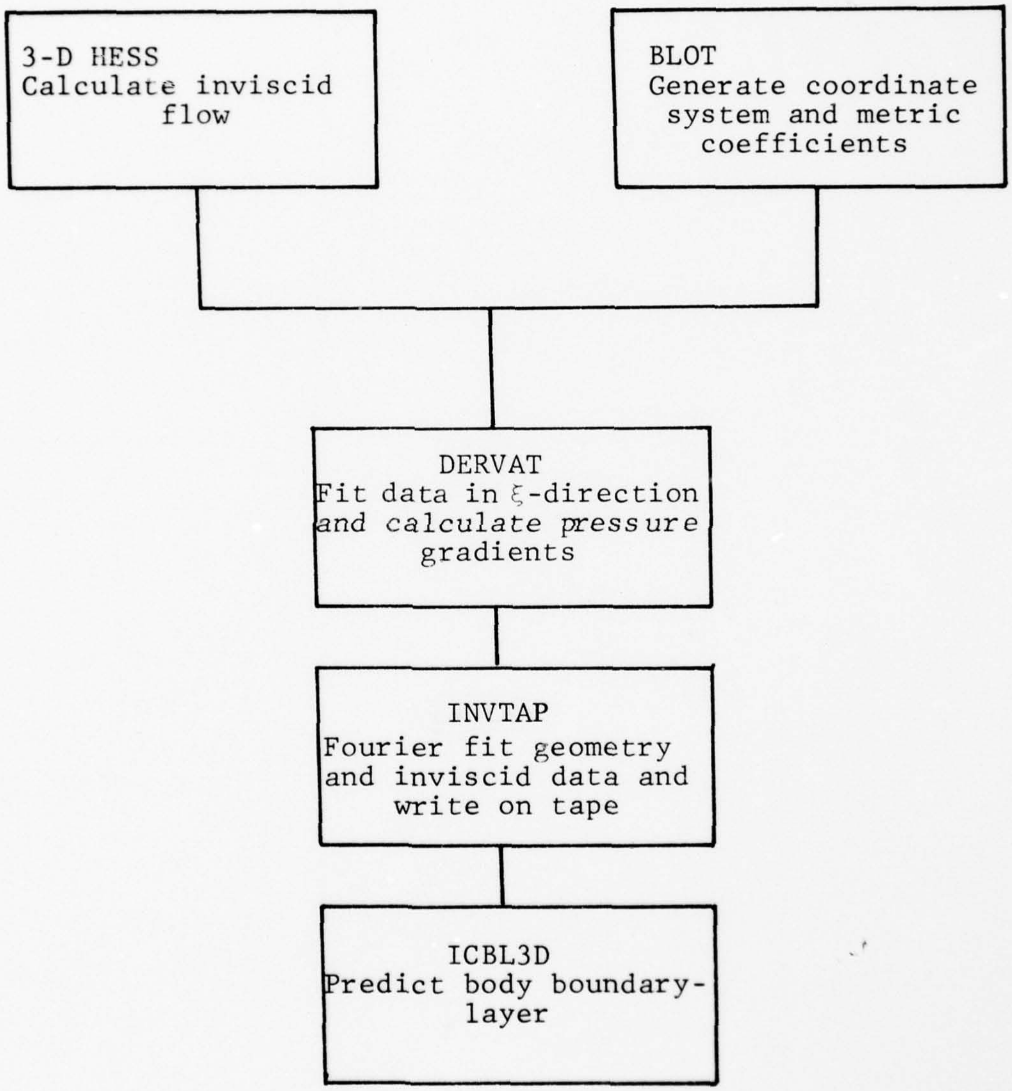

3-D HESS
Calculate inviscid flow
BLOT
Generate coordinate system and metric coefficients
DERVAT
Fit data in ξ-direction and calculate pressure gradients
INVTAP
Fourier fit geometry and inviscid data and write on tape
ICBL3D
Predict body boundary-layer

Figure 2b. BLOCK DIAGRAM FOR ICBL3D NONAXISYMMETRIC BODIES

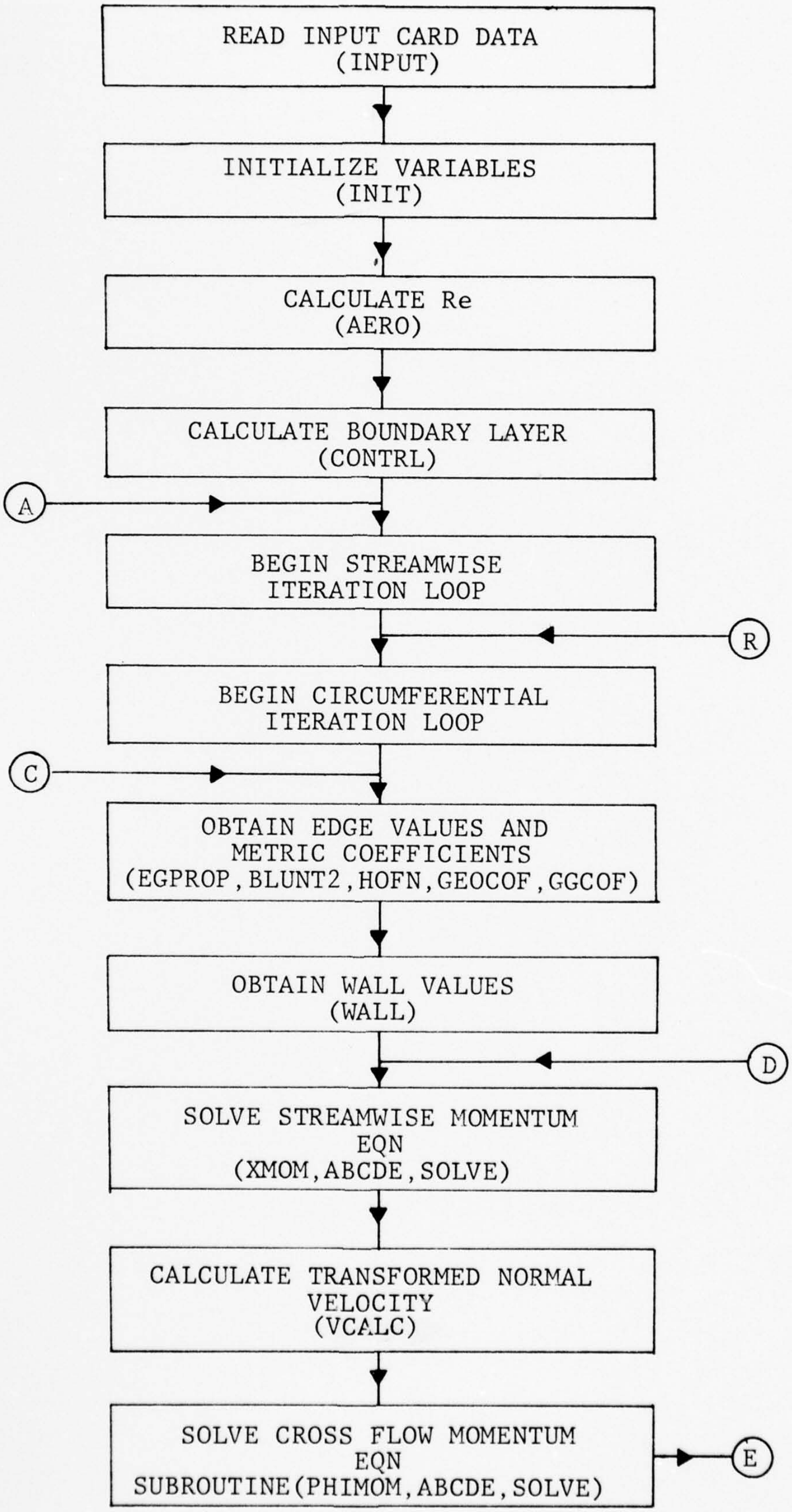


Figure 3. FLOW CHART OF PROGRAM ICBL3D

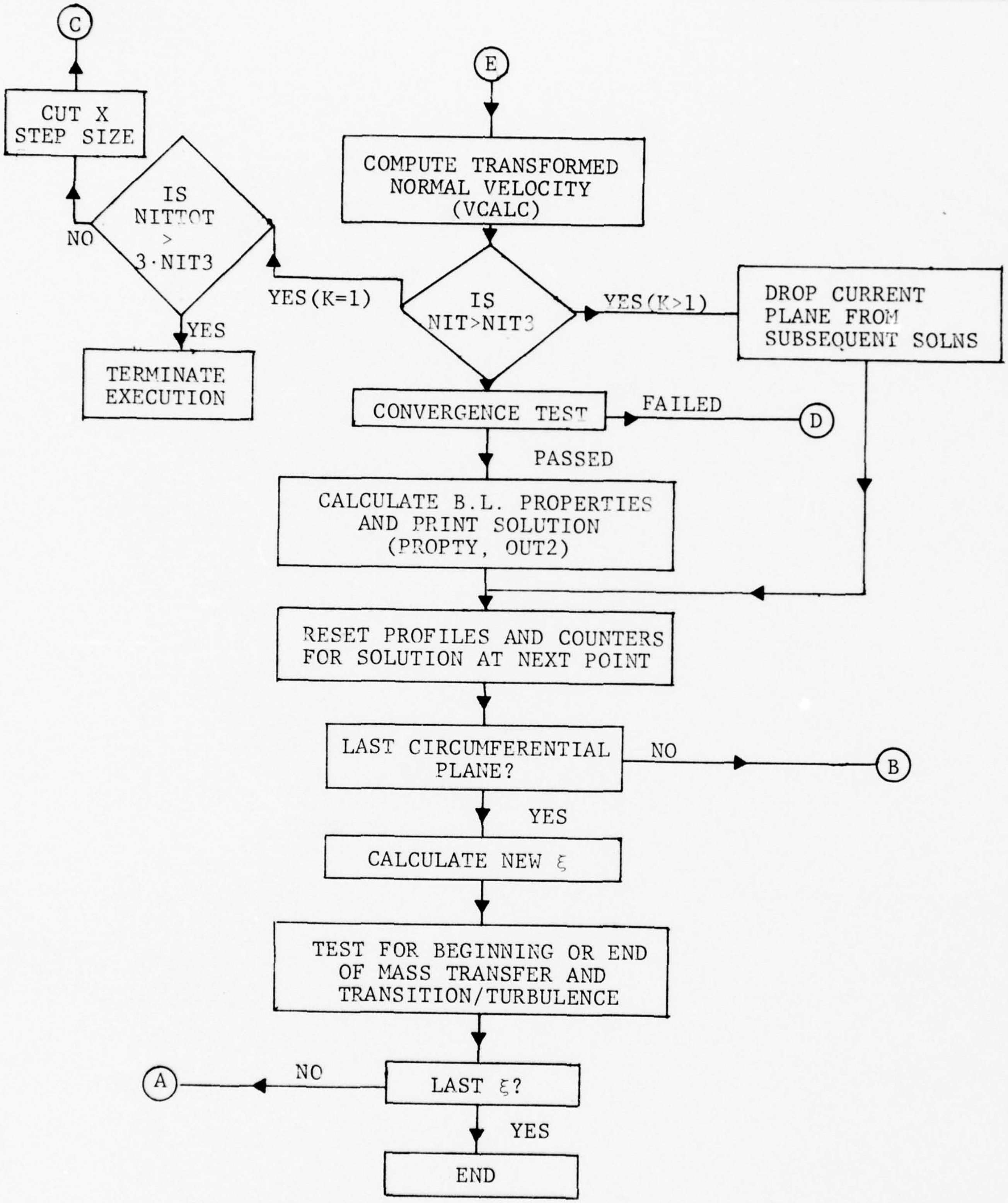

C
CUT X STEP SIZE
IS NITTOT > 3·NIT3
NO
YES
TERMINATE EXECUTION
YES(K=1)
E
COMPUTE TRANSFORMED NORMAL VELOCITY (VCALC)
IS NIT>NIT3
YES(K>1)
DROP CURRENT PLANE FROM SUBSEQUENT SOLNS
CONVERGENCE TEST
FAILED
D
PASSED
CALCULATE B.L. PROPERTIES AND PRINT SOLUTION (PROPTY, OUT2)
RESET PROFILES AND COUNTERS FOR SOLUTION AT NEXT POINT
LAST CIRCUMFERENTIAL PLANE?
NO
B
YES
CALCULATE NEW ξ
TEST FOR BEGINNING OR END OF MASS TRANSFER AND TRANSITION/TURBULENCE
A
NO
LAST ξ?
YES
END

Figure 3 continued: FLOW CHART OF PROGRAM ICBL3D
( ) = Subroutine Names

REFERENCES

1. Dwoyer, D. L., Lewis, C. H. and Gogineni, P. R., "Three-Dimensional Incompressible Boundary Layers on Blunt Bodies Including Effects of Turbulence, Surface Curvature and Heat and Mass Transfer. Part I: Analysis and Results." VPI&SU Aero-063, May 1977.

2. Blottner, F. G. and Ellis, M., "Three-Dimensional Incompressible Boundary Layer on Blunt Bodies." Sandia Laboratories Report No. SLA-73-0366, Albuquerque, New Mexico, April 1973.

3. Hess, J. L. and Martin, R. P., Jr., "Improved Solution for Potential Flow about Arbitrary Axisymmetric Bodies by the Use of a Higher-Order Surface Source Method. Part I. Theory and Results." NASA Contractor Report No. NASA CR 134694, NASA Lewis Research Center, Cleveland, Ohio, July 1974.

4. Hess, J. L. and Smith, A.M.O., "Calculation of Non-Lifting Potential Flow about Arbitrary Three-Dimensional Bodies." March 1962, McDonnell-Douglas Report No. ES 40622.

APPENDIX I

Job Control Language for Program TAPGEN

```
DATE   05/17/77


// JOB CARD
/*PRIORITY URGENT
/*JOBPARM LINES=10,CARDS=0
//STEP1 EXEC FORTGCG,TIME=4,REGION=520K,PARM.FORT='NOSOURCE'
//FORT.SYSIN DD *

               PROGRAM HESS    FORTRAN SOURCE DECK

//GO.FT05F001 DD *

            INPUT DATA DECK FOR  PROGRAM  HESS

//FT06F001  DD  SYSOUT=A
//FT07F001  DD  DUMMY
//FT01F001  DD  UNIT=SYSDA,DISP=(NEW,DELETE),SPACE=(TRK,(50,1))
//FT02F001  DD  UNIT=SYSDA,DISP=(NEW,DELETE),SPACE=(TRK,(50,1))
//FT03F001  DD  UNIT=SYSDA,DISP=(NEW,DELETE),SPACE=(TRK,(50,1))
//FT04F001  DD  UNIT=SYSDA,DISP=(NEW,DELETE),SPACE=(TRK,(50,1))
//FT08F001  DD  UNIT=SYSDA,DISP=(NEW,DELETE),SPACE=(TRK,(50,1))
//FT09F001  DD  UNIT=SYSDA,DISP=(NEW,DELETE),SPACE=(TRK,(50,1))
//FT10F001  DD  UNIT=SYSDA,DISP=(NEW,DELETE),SPACE=(TRK,(50,1))
//FT11F001  DD  UNIT=SYSDA,DISP=(NEW,DELETE),SPACE=(TRK,(50,1))
//FT12F001  DD  UNIT=SYSDA,DISP=(NEW,DELETE),SPACE=(TRK,(50,1))
//FT13F001  DD  UNIT=SYSDA,DISP=(NEW,DELETE),SPACE=(TRK,(50,1))
//FT15F001  DD  UNIT=SYSDA,DISP=(NEW,DELETE),SPACE=(TRK,(50,1))
//FT16F001  DD  UNIT=SYSDA,DISP=(NEW,PASS),SPACE=(TRK,(50,1)),DSN=&&T1
/*
//STEP2 EXEC FORTGCG,COND=(0,NE),TIME=1,REGION=300K
//FORT.SYSIN DD *

            PROGRAM AOAT     FORTRAN SOURCE DECK

//GO.FT05F001 DD *

            INPUT DATA DECK FOR  PROGRAM  AOAT

 //FT06F001 DD DUMMY
//FT16F001 DD UNIT=SYSDA,DISP=(OLD,DELETE),DSN=&&T1
//FT22F001 DD DUMMY
//FT30F001 DD DUMMY
//FT10F001 DD UNIT=SYSDA,DISP=(NEW,PASS),SPACE=(TRK,(50,1)),DSN=&&T2
/*
//STEP3 EXEC FORTGCG,COND=(0,NE),TIME=10,REGION=500K
//FORT.SYSIN DD *

            PROGRAM BLOT     FORTRAN SOURCE DECK

//GO.FT05F001 DD DUMMY

            INPUT DATA DECK FOR  PROGRAM  BLOT
```

```
DATE  05/17/77


//FT06F001 DD DUMMY
//FT08F001 DD DUMMY
//FT11F001 DD DUMMY
//FT12F001 DD UNIT=SYSDA,DISP=(NEW,DELETE),SPACE=(TRK,(50,1))
//FT13F001 DD UNIT=SYSDA,DISP=(NEW,DELETE),SPACE=(TRK,(50,1)),
//            DCB=(RECFM=VS,LRECL=2884,BLKSIZE=2888)
//FT10F001 DD UNIT=SYSDA,DISP=(OLD,DELETE),DSN=&&T2
//FT25F001 DD UNIT=SYSDA,DISP=(NEW,CATLG,CATLG),SPACE=(TRK,(50,1)),
//            DSN=AF01F3.T.T4
//FT04F001 DD *
/*
//STEP4 EXEC FORTGCG,TIME=4,REGION=500K
//FORT.SYSIN DD *

        PROGRAM DERVAT   FORTRAN SOURCE DECK

//FT06F001 DD SYSOUT=A
//FT25F001 DD UNIT=SYSDA,DISP=(OLD,KEEP),DSN=AF01F3.T.T4
//FT26F001 DD UNIT=SYSDA,DISP=(NEW,PASS),SPACE=(TRK,(50,1)),DSN=&&T4
//FT29F001 DD SYSOUT=A
//FT30F001 DD SYSOUT=A
//STEP5 EXEC FORTGCG,COND=(0,NE),TIME=4,REGION=300K
//FORT.SYSIN DD *

         PROGRAM INVTAP    FORTRAN SOURCE DECK

//GO.FT05F001 DD DUMMY
//FT06F001 DD SYSOUT=A
//FT10F001 DD UNIT=SYSDA,DISP=(NEW,CATLG),SPACE=(TRK,(50),RLSE),
//            DSN=A505F3.INVTAP.EL412,VOL=SER=USERPK,
// DCB=(RECFM=VS,LRECL=117,BLKSIZE=121)
//FT25F001 DD UNIT=SYSDA,DISP=(OLD,DELETE),DSN=&&T4
//FT30F001 DD SYSOUT=A
/*
//
```

APPENDIX II

TAPGEN Sample Input

```
DATE  04/21/77

         INPUT DATA DECK FOR  PROGRAM  HESS                                     1
          ELLIPSOID (4:1)  AT   ALPHA= 10 DEG   120 ELEMENTS        2           2
1011001   00 00 0              1                                                3
01001  121   1.0         1.0                                                    4
         1          0          1     1.0     0.25                               5
```

```
DATE  04/21/77


          INPUT DATA DECK FOR  PROGRAM  BLOT
TITLE   CASE 7  ELLIPSOID(4:1) - 10 DEG  ANGLE OF ATTACK
LIMTS    31   26   01   10    1     0  6.00        100.0
DW       .02          .02          .02          .02          .02          .02          .02
DW       .02          .02          .02          .02          .02          .02          .02
DW       .02          .02          .02          .02          .02          .02          .02
DW       .02          .02          .02          .02          .02          .02          .02
DN       .3           .3           .3           .3           .3           .3           .3
DN       .3           .3           .3           .3           .3           .3           .3
DN       .3           .3           .3           .3           .3           .3           .3
DN       .3           .3           .3           .3           .3           .3           .3
DN       .3           .3           .3           .3           .3           .3           .3
DPSI     .01
SIZE    4.00          .25          .25      .0000001 +10.0         1.0          0.5
```

APPENDIX III

TAPGEN Sample Output

```
                         INVISCID EDGE CONDITIONS FOR BOUNDARY LAYER SOLUTION
                              TAKEN FROM THE INVISCID FLOW FIELD DATA


   NUMBER OF PLANES IN THE INVISCID DATA = 26

   ALPHA=  0.0

   UNIFORM FLOW STARTING SOLUTION FOR THE INVISCID FLOW FIELD
WALL DATA AT STATION    1, XI = .0

K   HPSI          HW            R             S             UPSI          UW            X

1   0.100000D 01  0.0           0.0           0.0           0.0           0.0           0.0
2   0.100000D 01  0.0           0.0           0.0           0.0           0.0           0.0
3   0.100000D 01  0.0           0.0           0.0           0.0           0.0           0.0
4   0.100000D 01  0.0           0.0           0.0           0.0           0.0           0.0
5   0.100000D 01  0.0           0.0           0.0           0.0           0.0           0.0
6   0.100000D 01  0.0           0.0           0.0           0.0           0.0           0.0
7   0.100000D 01  0.0           0.0           0.0           0.0           0.0           0.0
8   0.100000D 01  0.0           0.0           0.0           0.0           0.0           0.0
9   0.100000D 01  0.0           0.0           0.0           0.0           0.0           0.0
10  0.100000D 01  0.0           0.0           0.0           0.0           0.0           0.0
11  0.100000D 01  0.0           0.0           0.0           0.0           0.0           0.0
12  0.100000D 01  0.0           0.0           0.0           0.0           0.0           0.0
13  0.100000D 01  0.0           0.0           0.0           0.0           0.0           0.0
14  0.100000D 01  0.0           0.0           0.0           0.0           0.0           0.0
15  0.100000D 01  0.0           0.0           0.0           0.0           0.0           0.0
16  0.100000D 01  0.0           0.0           0.0           0.0           0.0           0.0
17  0.100000D 01  0.0           0.0           0.0           0.0           0.0           0.0
18  0.100000D 01  0.0           0.0           0.0           0.0           0.0           0.0
19  0.100000D 01  0.0           0.0           0.0           0.0           0.0           0.0
20  0.100000D 01  0.0           0.0           0.0           0.0           0.0           0.0
21  0.100000D 01  0.0           0.0           0.0           0.0           0.0           0.0
22  0.100000D 01  0.0           0.0           0.0           0.0           0.0           0.0
23  0.100000D 01  0.0           0.0           0.0           0.0           0.0           0.0
24  0.100000D 01  0.0           0.0           0.0           0.0           0.0           0.0
25  0.100000D 01  0.0           0.0           0.0           0.0           0.0           0.0
26  0.100000D 01  0.0           0.0           0.0           0.0           0.0           0.0
K   D(UPSI)       D(UW)           D(R)          D2(R)           D(HPSI)         D(HW)         D(X)

1   0.468502D 01   0.0            0.101447D 01  -0.480908D 00  -0.666134D-15   0.637409D 01  -0.139029D-01
2   0.474720D 01   0.462593D-16   0.101447D 01  -0.480908D 00   0.422081D-07   0.637409D 01  -0.139029D-01
3   0.477582D 01   0.481097D-16   0.101447D 01  -0.480908D 00   0.422081D-07   0.637409D 01  -0.139029D-01
4   0.478332D 01   0.888178D-16   0.101447D 01  -0.480908D 00   0.422081D-07   0.637409D 01  -0.139029D-01
5   0.478518D 01  -0.100660D-14   0.101447D 01  -0.480908D 00   0.422082D-07   0.637409D 01  -0.139029D-01
6   0.478563D 01   0.473695D-15   0.101447D 01  -0.480908D 00   0.422081D-07   0.637409D 01  -0.139029D-01
7   0.478575D 01  -0.296059D-14   0.101447D 01  -0.480908D 00   0.422081D-07   0.637409D 01  -0.139029D-01
8   0.478577D 01   0.0            0.101447D 01  -0.480908D 00   0.422081D-07   0.637409D 01  -0.139029D-01
9   0.478578D 01  -0.115463D-14   0.101447D 01  -0.480908D 00   0.422081D-07   0.637409D 01  -0.139029D-01
10  0.478578D 01  -0.947390D-15   0.101447D 01  -0.480908D 00   0.422081D-07   0.637409D 01  -0.139029D-01
11  0.478578D 01  -0.375625D-15   0.101447D 01  -0.480908D 00   0.422081D-07   0.637409D 01  -0.139029D-01
12  0.478578D 01  -0.249800D-15   0.101447D 01  -0.480908D 00   0.422081D-07   0.637409D 01  -0.139029D-01
13  0.478578D 01  -0.148030D-15   0.101447D 01  -0.480908D 00   0.422081D-07   0.637409D 01  -0.139029D-01
14  0.478578D 01  -0.555112D-16   0.101447D 01  -0.480908D 00   0.422081D-07   0.637409D 01  -0.139029D-01
15  0.478578D 01   0.666134D-16   0.101447D 01  -0.480908D 00   0.422082D-07   0.637409D 01  -0.139029D-01
16  0.478578D 01   0.131561D-14   0.101447D 01  -0.480908D 00   0.422082D-07   0.637409D 01  -0.139029D-01
17  0.478578D 01  -0.189478D-14   0.101447D 01  -0.480908D 00   0.422082D-07   0.637409D 01  -0.139029D-01
```

```
 18   0.478578D 01   0.0             0.101447D 01  -0.480908D 00   0.422082D-07   0.637409D 01  -0.139029D-01
 19   0.478578D 01   0.142109D-14    0.101447D 01  -0.480908D 00   0.422082D-07   0.637409D 01  -0.139029D-01
 20   0.478578D 01   0.142109D-14    0.101447D 01  -0.480908D 00   0.422082D-07   0.637409D 01  -0.139029D-01
 21   0.478578D 01   0.153951D-14    0.101447D 01  -0.480908D 00   0.422082D-07   0.637409D 01  -0.139029D-01
 22   0.478578D 01  -0.621725D-15    0.101447D 01  -0.480908D 00   0.422082D-07   0.637409D 01  -0.139029D-01
 23   0.478578D 01   0.207242D-15    0.101447D 01  -0.480908D 00   0.422082D-07   0.637409D 01  -0.139029D-01
 24   0.478578D 01  -0.740149D-16    0.101447D 01  -0.480908D 00   0.422082D-07   0.637409D 01  -0.139029D-01
 25   0.478578D 01  -0.259052D-16    0.101447D 01  -0.480908D 00   0.422082D-07   0.637409D 01  -0.139029D-01
 26   0.478578D 01   0.225875D-20    0.101447D 01  -0.480908D 00   0.422082D-07   0.637409D 01  -0.139029D-01

                                FOURIER COEFFICIENTS

  K    AXSRFS          ARADS          AHXS           AHWS           AUS            AVS

  1   0.0             0.0             0.100000D 01   0.0            0.0            0.0
  2   0.0             0.0             0.710543D-16   0.0            0.0            0.0
  3   0.0             0.0            -0.532907D-16   0.0            0.0            0.0
  4   0.0             0.0             0.124345D-15   0.0            0.0            0.0
  5   0.0             0.0            -0.266454D-15   0.0            0.0            0.0
  6   0.0             0.0             0.581757D-15   0.0            0.0            0.0
  7   0.0             0.0            -0.763833D-15   0.0            0.0            0.0
  8   0.0             0.0             0.666134D-17   0.0            0.0            0.0
  9   0.0             0.0            -0.941469D-15   0.0            0.0            0.0
 10   0.0             0.0             0.532907D-15   0.0            0.0            0.0
 11   0.0             0.0            -0.177636D-15   0.0            0.0            0.0
 12   0.0             0.0             0.710543D-15   0.0            0.0            0.0
 13   0.0             0.0            -0.408562D-15   0.0            0.0            0.0
 14   0.0             0.0             0.279776D-15   0.0            0.0            0.0
 15   0.0             0.0            -0.337508D-15   0.0            0.0            0.0
 16   0.0             0.0             0.864864D-15   0.0            0.0            0.0
 17   0.0             0.0            -0.177636D-15   0.0            0.0            0.0
 18   0.0             0.0             0.932587D-16   0.0            0.0            0.0
 19   0.0             0.0             0.570655D-15   0.0            0.0            0.0
 20   0.0             0.0             0.544009D-16   0.0            0.0            0.0
 21   0.0             0.0             0.954792D-16   0.0            0.0            0.0
 22   0.0             0.0             0.633937D-15   0.0            0.0            0.0
 23   0.0             0.0            -0.248690D-15   0.0            0.0            0.0
 24   0.0             0.0             0.308642D-15   0.0            0.0            0.0
 25   0.0             0.0             0.802691D-15   0.0            0.0            0.0
 26   0.0             0.0             0.0            0.0            0.0            0.0

  K    AUSX           AVSX           ARADSX         ARDSXX         AHYSX          AHWSX          AX        A??

  1   0.478169D 01  -0.199247D-15   0.101447D 01  -0.480908D 00   0.41364 D-07   0.637409D 01   0.0      -0.139029D-01
  2  -0.810240D-02  -0.683902D-15   0.106581D-15   0.169420D-14  -0.168834D-08  -0.435207D-15   0.0       0.114769D-15
  3  -0.789167D-02   0.327840D-16  -0.177636D-16   0.170197D-14  -0.168832D-08   0.550671D-15   0.0       0.317851D-16
  4  -0.756208D-02   0.660848D-16   0.158762D-15  -0.470735D-15  -0.168832D-08   0.183853D-14   0.0      -0.164452D-16
  5  -0.714123D-02   0.106609D-15  -0.301981D-15   0.541234D-14  -0.168833D-08  -0.246914D-14   0.0       0.451026D-17
  6  -0.665997D-02   0.398154D-15   0.655032D-15  -0.597300D-15  -0.168832D-08   0.446754D-14   0.0       0.224820D-16
  7  -0.614783D-02  -0.232429D-15  -0.746070D-15   0.122236D-14  -0.168833D-08  -0.508038D-14   0.0       0.448710D-16
  8  -0.562997D-02   0.172216D-15   0.155431D-16   0.347500D-15  -0.168832D-08  -0.275335D-15   0.0       0.777156D-17
  9  -0.512573D-02   0.119622D-15  -0.976996D-15   0.395350D-14  -0.168833D-08  -0.691003D-14   0.0       0.145703D-16
 10  -0.464859D-02  -0.400329D-15   0.573935D-15  -0.161992D-14  -0.168833D-08   0.356160D-14   0.0       0.935323D-17
 11  -0.420688D-02   0.222658D-15  -0.177636D-15  -0.979217D-15  -0.168832D-08  -0.179412D-14   0.0       0.269352D-16
 12  -0.380488D-02  -0.297794D-16   0.691669D-15   0.432987D-16  -0.168832D-08   0.389910D-14   0.0       0.261943D-17
 13  -0.344394D-02  -0.238665D-15  -0.373035D-15   0.122569D-14  -0.168832D-08  -0.307310D-14   0.0       0.471845D-17
 14  -0.312340D-02   0.293416D-15   0.280886D-15  -0.197981D-14  -0.168832D-08   0.191402D-14   0.0       0.227249D-17
 15  -0.284140D-02   0.206496D-15  -0.301981D-15  -0.214273D-14  -0.168832D-08  -0.280664D-14   0.0       0.219963D-16
 16  -0.259544D-02  -0.618007D-16   0.872635D-15   0.111577D-14  -0.168832D-08   0.491163D-14   0.0      -0.152656D-17
 17  -0.238272D-02  -0.142117D-15  -0.142109D-15  -0.566214D-16  -0.168833D-08  -0.131450D-14   0.0      -0.277556D-17
 18  -0.220046D-02  -0.367406D-17   0.117684D-15  -0.839329D-15  -0.168833D-08   0.701661D-15   0.0      -0.721645D-17
 19  -0.204602D-02  -0.485170D-15   0.621725D-15  -0.239142D-14  -0.168833D-08   0.317968D-14   0.0      -0.201228D-17
 20  -0.191702D-02  -0.234965D-15  -0.355271D-16   0.327738D-14  -0.168833D-08  -0.133227D-15   0.0       0.444089D-17
 21  -0.181136D-02   0.441967D-15   0.142109D-15  -0.788258D-15  -0.168833D-08   0.728306D-15   0.0      -0.137553D-16
 22  -0.172729D-02   0.791287D-16   0.622835D-15  -0.175415D-15  -0.168833D-08   0.384581D-14   0.0      -0.235922D-16
 23  -0.166337D-02   0.376199D-15  -0.195399D-15  -0.184408D-14  -0.168832D-08  -0.150990D-14   0.0      -0.134778D-17
 24  -0.161849D-02   0.369650D-15   0.245359D-15   0.382361D-14  -0.168832D-08   0.190958D-14   0.0      -0.381639D-17
```

```
26  -0.791528D-03   0.0             0.0                0.102141D-15  -0.344161D-09  -0.222045D-16   0.0                    -0.943860D-17
WALL DATA AT STATION   2, XI = .1000D 00

K    HPSI            HW              R               S               UPSI            UW              X

 1   0.100000D 01    0.613045D 00    0.975691D-01    0.100000D 00    0.394865D 00    0.0             0.190850D-01
 2   0.100000D 01    0.613045D 00    0.975691D-01    0.100000D 00    0.393903D 00    0.444089D-17    0.190850D-01
 3   0.100000D 01    0.613045D 00    0.975691D-01    0.100000D 00    0.393495D 00   -0.444089D-17    0.190850D-01
 4   0.100000D 01    0.613045D 00    0.975691D-01    0.100000D 00    0.393384D 00    0.222045D-17    0.190850D-01
 5   0.100000D 01    0.613045D 00    0.975691D-01    0.100000D 00    0.393362D 00   -0.888178D-17    0.190850D-01
 6   0.100000D 01    0.613045D 00    0.975691D-01    0.100000D 00    0.393355D 00    0.0             0.190850D-01
 7   0.100000D 01    0.613045D 00    0.975691D-01    0.100000D 00    0.393354D 00   -0.355271D-16    0.190850D-01
 8   0.100000D 01    0.613045D 00    0.975691D-01    0.100000D 00    0.393353D 00    0.0             0.190850D-01
 9   0.100000D 01    0.613045D 00    0.975691D-01    0.100000D 00    0.393353D 00    0.0             0.190850D-01
10   0.100000D 01    0.613045D 00    0.975691D-01    0.100000D 00    0.393353D 00   -0.666134D-17    0.190850D-01
11   0.100000D 01    0.613045D 00    0.975691D-01    0.100000D 00    0.393353D 00   -0.222045D-17    0.190850D-01
12   0.100000D 01    0.613045D 00    0.975691D-01    0.100000D 00    0.393353D 00   -0.222045D-17    0.190850D-01
13   0.100000D 01    0.613045D 00    0.975691D-01    0.100000D 00    0.393353D 00   -0.444089D-17    0.190850D-01
14   0.100000D 01    0.613045D 00    0.975691D-01    0.100000D 00    0.393353D 00    0.0             0.190850D-01
15   0.100000D 01    0.613045D 00    0.975691D-01    0.100000D 00    0.393353D 00    0.666134D-17    0.190850D-01
16   0.100000D 01    0.613045D 00    0.975691D-01    0.100000D 00    0.393353D 00    0.111022D-16    0.190850D-01
17   0.100000D 01    0.613045D 00    0.975691D-01    0.100000D 00    0.393353D 00   -0.111022D-16    0.190850D-01
18   0.100000D 01    0.613045D 00    0.975691D-01    0.100000D 00    0.393353D 00    0.0             0.190850D-01
19   0.100000D 01    0.613045D 00    0.975691D-01    0.100000D 00    0.393353D 00    0.355271D-16    0.190850D-01
20   0.100000D 01    0.613045D 00    0.975691D-01    0.100000D 00    0.393353D 00    0.355271D-16    0.190850D-01
21   0.100000D 01    0.613045D 00    0.975691D-01    0.100000D 00    0.393353D 00    0.0             0.190850D-01
22   0.100000D 01    0.613045D 00    0.975691D-01    0.100000D 00    0.393353D 00    0.133227D-16    0.190850D-01
23   0.100000D 01    0.613045D 00    0.975691D-01    0.100000D 00    0.393353D 00   -0.666134D-17    0.190850D-01
24   0.100000D 01    0.613045D 00    0.975691D-01    0.100000D 00    0.393353D 00    0.222045D-17    0.190850D-01
25   0.100000D 01    0.613045D 00    0.975691D-01    0.100000D 00    0.393353D 00   -0.222045D-17    0.190850D-01
26   0.100000D 01    0.613045D 00    0.975691D-01    0.100000D 00    0.393353D 00    0.677626D-22    0.190850D-01

K    D(UPSI)         D(UW)           D(R)            D2(R)           D(HPSI)         D(HW)           D(X)

 1   0.321743D 01    0.0             0.927255D 00   -0.115247D 01    0.247025D-14    0.582612D 01    0.366949D 00
 2   0.317523D 01    0.203541D-16    0.927255D 00   -0.115247D 01   -0.418395D-07    0.582612D 01    0.366949D 00
 3   0.315617D 01   -0.814164D-16    0.927255D 00   -0.115247D 01   -0.418395D-07    0.582612D 01    0.366949D 00
 4   0.315117D 01   -0.185037D-16    0.927255D 00   -0.115247D 01   -0.418395D-07    0.582612D 01    0.366949D 00
 5   0.314993D 01    0.310862D-15    0.927255D 00   -0.115247D 01   -0.418395D-07    0.582612D 01    0.366949D 00
 6   0.314963D 01   -0.177636D-15    0.927255D 00   -0.115247D 01   -0.418395D-07    0.582612D 01    0.366949D 00
 7   0.314955D 01   -0.100660D-14    0.927255D 00   -0.115247D 01   -0.418395D-07    0.582612D 01    0.366949D 00
 8   0.314953D 01    0.0             0.927255D 00   -0.115247D 01   -0.418395D-07    0.582612D 01    0.366949D 00
 9   0.314953D 01    0.562513D-15    0.927255D 00   -0.115247D 01   -0.418395D-07    0.582612D 01    0.366949D 00
10   0.314953D 01    0.440388D-15    0.927255D 00   -0.115247D 01   -0.418395D-07    0.582612D 01    0.366949D 00
11   0.314953D 01    0.227596D-15    0.927255D 00   -0.115247D 01   -0.418395D-07    0.582612D 01    0.366949D 00
12   0.314953D 01    0.869675D-16    0.927255D 00   -0.115247D 01   -0.418395D-07    0.582612D 01    0.366949D 00
13   0.314953D 01    0.259052D-16    0.927255D 00   -0.115247D 01   -0.418395D-07    0.582612D 01    0.366949D 00
14   0.314953D 01    0.370074D-16    0.927255D 00   -0.115247D 01   -0.418395D-07    0.582612D 01    0.366949D 00
15   0.314953D 01    0.259052D-16    0.927255D 00   -0.115247D 01   -0.418395D-07    0.582612D 01    0.366949D 00
16   0.314953D 01   -0.534757D-15    0.927255D 00   -0.115247D 01   -0.418395D-07    0.582612D 01    0.366949D 00
17   0.314953D 01    0.773455D-15    0.927255D 00   -0.115247D 01   -0.418395D-07    0.582612D 01    0.366949D 00
18   0.314953D 01    0.0             0.927255D 00   -0.115247D 01   -0.418395D-07    0.582612D 01    0.366949D 00
19   0.314953D 01   -0.296059D-15    0.927255D 00   -0.115247D 01   -0.418395D-07    0.582612D 01    0.366949D 00
20   0.314953D 01   -0.296059D-15    0.927255D 00   -0.115247D 01   -0.418395D-07    0.582612D 01    0.366949D 00
21   0.314953D 01   -0.710543D-15    0.927255D 00   -0.115247D 01   -0.418395D-07    0.582612D 01    0.366949D 00
22   0.314953D 01    0.451491D-15    0.927255D 00   -0.115247D 01   -0.418395D-07    0.582612D 01    0.366949D 00
23   0.314953D 01   -0.122125D-15    0.927255D 00   -0.115247D 01   -0.418395D-07    0.582612D 01    0.366949D 00
24   0.314953D 01    0.407082D-16    0.927255D 00   -0.115247D 01   -0.418395D-07    0.582612D 01    0.366949D 00
25   0.314953D 01   -0.222045D-16    0.927255D 00   -0.115247D 01   -0.418395D-07    0.582612D 01    0.366949D 00
26   0.314953D 01   -0.395282D-21    0.927255D 00   -0.115247D 01   -0.418395D-07    0.582612D 01    0.366949D 00
                        FOURIER COEFFICIENTS

K    AXSRFS          ARADS           AHXS            AHWS            AUS             AVS
```

APPENDIX IV

Program TAPGEN Data FORMAT Sheets

IBM FORTRAN Coding Form

GX28-7327-6 U/M 050**
Printed in U.S.A.

PROGRAM HESS

PROGRAMMER DATE

PUNCHING INSTRUCTIONS GRAPHIC PUNCH

PAGE OF

CARD ELECTRO NUMBER*

STATEMENT NUMBER | CONT. | FORTRAN STATEMENT | IDENTIFICATION SEQUENCE

```
DATA DECK

CARD 1 - HEADER CARD
                 HEADER                                         CASE
[                                                          ]   [    ]
               ( 15A4 )                                         (I6)
        HEADER    ANY ALPHANUMERIC RUN DESCRIPTION
        CASE      CASE NUMBER
```

Number of forms per pad may vary slightly

IBM FORTRAN Coding Form

GX28-7327-6 U/M 050**
Printed in U.S.A.

PROGRAM HESS
PROGRAMMER
DATE
PUNCHING INSTRUCTIONS GRAPHIC PUNCH
PAGE OF
CARD ELECTRO NUMBER*

STATEMENT NUMBER — FORTRAN STATEMENT — IDENTIFICATION SEQUENCE

```
CARD 2 - CONTROL FLAG CARD

NB NNU IAXI ICRØSS IØFF IØNLY IELPSE   IPRTRB IPØTNL*   IPTANV IVØRT IØMITA IOMITC ISURFV IPRSCV IALLV IEXCRS* IGENBC IRNGW   IPNCH IUNIT   IVIJ ICØEF IPRINT IRAKF

 (7I1 ,3X,2I1,1X,10I1,4X,I1,I2,4I1)
      NB       NUMBER OF BODIES (1≤NB≤9)
      NNU      NUMBER OF NONUNIFORM FLOWS (0≤NNU≤5)
      IAXI     AXISYMMETRIC FLOW FLAG
      ICRØSS   CROSSFLOW FLAG
      IØFF     OFF BODY POINT INPUT FLAG
      IØNLY    BASIC-DATA-ONLY FLAG
      IELPSE   ELLIPSE GENERATOR FLAG (SEE ALSO CARD 5).
      IPRTRB   PERTURBATION VELOCITIES ONLY
      IPØTNL*  SOLVE POTENTIAL MATRIX
      IPTANV   PRESCRIBED TANGENTIAL VELOCITY (FOR LAST IPTANV BODIES)
      IVØRT    STRIP-RING VORTICITY FLAG
      IØMITA   OMIT AXISYMMETRIC UNIFORM FLOW SOLUTION
      IOMITC   OMIT CROSSFLOW UNIFORM FLOW SOLUTION
      ISURFV   SURFACE VORTICITY (INSTEAD OF SOURCES) FOR FINAL ISURFV BODIES
      IPRSCV   PRESCRIBED VALUES OF SURFACE VORTEX STRENGTHS FOR FINAL ISURFV
      IALLV    ALL BODIES ARE SURFACE VORTICITY BODIES            WILL BE INPUT.
      IEXCRS*  EXTRA CROSSFLOW
```

*Number of forms per pad may vary slightly

*Available if and only if NØNEWF = 1, ISIGF = 2, IGEØMF = 1.

IBM

FORTRAN Coding Form

GX28-7327-6 U/M 050**
Printed in U.S.A.

PROGRAM	HESS	PUNCHING INSTRUCTIONS	GRAPHIC
PROGRAMMER	DATE		PUNCH

PAGE OF

CARD ELECTRO NUMBER*

STATEMENT NUMBER | CONT. | FORTRAN STATEMENT | IDENTIFICATION SEQUENCE

```
CARD 2IGENBC    GENERATED BOUNDARY CONDITIONS
      IRNGW     RING WING OPTION
      IPNCH     PUNCHED OUTPUT
      IUNIT     UNIT NUMBER FOR INPUT COORDINATES (default = 05)
      IVIJ      MATRIX PRINT FLAG
      ICØEF     MATRIX-ASSEMBLY COEFFICIENT PRINT FLAG
      IPRINT    VERY DETAILED MATRIX CONSTRUCTION PRINT FLAG
      IRAKF     AUTOMATIC RAKE GENERATION FLAG
```

*Number of forms per pad may vary slightly

IBM FORTRAN Coding Form

GX28-7327-6 U/M 050**
Printed in U.S.A.

PROGRAM HESS

PROGRAMMER

DATE

PUNCHING INSTRUCTIONS GRAPHIC PUNCH

PAGE OF

CARD ELECTRO NUMBER*

STATEMENT NUMBER | CONT | FORTRAN STATEMENT | IDENTIFICATION SEQUENCE

```
CARD 3  - CHORD/MACH NUMBER CARD *

    CHØRD       XMACH
|          |          |
         2F10.6
         CHORD      REFERENCE CHORD LENGTH ( DEFAULT = 1.0)
         XMACH      MACH NUMBER FOR GOETHERT CORRECTION (0.0 IMPLIES INCOMPRESSIBLE)


 *THIS CARD USUALLY LEFT BLANK
```

*Number of forms per pad may vary slightly

IBM FORTRAN Coding Form

GX28-7327-6 U/M 050**
Printed in U.S.A.

PROGRAM HESS

PROGRAMMER DATE

PUNCHING INSTRUCTIONS GRAPHIC PUNCH

PAGE OF

CARD ELECTRO NUMBER*

FORTRAN STATEMENT

IDENTIFICATION SEQUENCE

CARD 4 BODY CONTROL CARD 1 OF 2

IGEØMF	ISIGF	ICURVN	NONEWF	IFORMT	NN	XMULT	YMULT	THETA	ADDX	ADDY

(5I1 ,1 I 5, 5F10. 4)

IGEØMF	0=CURVED ELEMENTS; 1=FLAT ELEMENTS
ISIGF	0=PARABOLIC σ; 1= LINEAR σ; 2= CONSTANT σ (ON EACH ELEMENT)
ICURVN	0=INTERNALLY CALCULATED ELEMENT CURVATURES
NØNEWF	0=USE NEWEST FORMULAE; 1=USE OLD FORMULAE (IMPLIES FLAT ELEMENTS AND CONSTANT σ)
IFØRMT	INPUT FORMAT FLAG
NN	NUMBER OF DEFINING ENDPOINTS FOR THIS BODY
XMULT	X-MULTIPLIER VALUE (DEFAULT = 1.0)
YMULT	Y-MULTIPLIER VALUE (DEFAULT = 1.0)
THETA	COORDINATE ROTATION VALUE (DEGREES, MEASURED ABOUT -Z-AXIS)
ADDX	X-INCREMENT (TO BE APPLIED TO ALL INPUT COORDINATES FOR THIS BODY)
ADDY	Y-INCREMENT (TO BE APPLIED TO ALL INPUT COORDINATES FOR THIS BODY)

*Number of forms per pad may vary slightly

IBM FORTRAN Coding Form

GX28-7327-6 U/M 050**
Printed in U.S.A.

PROGRAM	HESS	PUNCHING INSTRUCTIONS	GRAPHIC
PROGRAMMER	DATE		PUNCH

PAGE OF

CARD ELECTRO NUMBER

FORTRAN STATEMENT

```
CARD 5  BODY CONTROL  CARD 2 OF 2

      IBDN      ISUBKS        NLF          A          B

                  (3I10, 2F10.3)

       IBDN      "BODY" NUMBER ( SEQUENTIAL FOR BODIES, ZERO FOR OFF BODY POINTS)
       ISUBKS    SUBCASE FLAG
       NLF       NON-LIFTING FLAG (FOR COMBINATION CASES, ONLY)
       A         SEMI-MAJOR AXIS FOR ELLIPSE CASES }
       B         SEMI-MINOR AXIS FOR ELLIPSE CASES } IF IELPSE≠0
```

IBM FORTRAN Coding Form GX28-7327-6 U/M 050**
Printed in U.S.A.

PROGRAM AOAT

PROGRAMMER

DATE

PUNCHING INSTRUCTIONS GRAPHIC PUNCH

PAGE OF

CARD ELECTRO NUMBER*

STATEMENT NUMBER | CONT | FORTRAN STATEMENT | IDENTIFICATION SEQUENCE

```
DATA SET

CARD 1

IPRT N KPL    ALPD

     (3I3,F10.6)


 IPRT  =1 WRITES THE  OUTPUT OF  PROGRAM HESS ON  UNIT 6; =0,  DOES NOT WRITE
 N         NUMBER OF ELEMENTS USED IN THE PROGRAM HESS
 KPL       NUMBER OF MERIDONIAL PLANES REQUIRED
 ALPD      ANGLE OF ATTACK IN DEGREES
```

**Number of forms per pad may vary slightly

IBM FORTRAN Coding Form

GX28-7327-6 U/M 050**
Printed in U.S.A.

THREE-DIMENSIONAL, INCOMPRESSIBLE BOUNDARY LAYER ON BLUNT BODIES,
PART II: COMPUTER CODE - USER'S GUIDE, Molly A. Ellis, F. G. Blottner
RE: SLA-73-0704

PAGE 1 OF 1

FORTRAN STATEMENT

```
  TITLE  PROBLEM IDENTIFICATION                                   (5X,9A8)
1 TITLE

  LIMTS  JMAX KMAXLITER IPRT KPRT  PSIMAX      UINF               (5X,5I5,2E10.0)
2 LIMTS

  DW     Δω's AROUND THE BODY                                     (5X,7E10.0)
3 DW(K)

  DN     Δη's ACROSS THE BOUNDARY LAYER                           (5X,7E10.0)
4 DN(J)

  DPSI   Δξ - STEPSIZE TO BE USED ALONG THE BODY                  (5X,E10.0)
5 DPSI

  SIZE      A         RB        RC        EPS       ALP     CAPTH  CRI (5X,7E10.0)
6 SIZE
```

JMAX	No. of pts. across boundary layer	A	a - length of axis of triaxial ellipsoid
KMAX	No. of pts. around body (in phi-direction)	RB	R_b - radius of curvature of nose; plane y'=0
LITER	No. of iterations for initial profile	RC	R_c - radius of curvature of nose; plane z'=0
IPRT	Every how many steps printing will occur along the body	EPS	Convergence criterion; body coordinates
KPRT	Every how many steps printing will occur around the body	ALP	α - angle of attack (deg)
PSIMAX	Distance along body at which program will terminate	CAPTH	Θ - indicates finite-difference scheme used
UINF	Freestream velocity - V_∞	CRI	1 Implicit scheme, ½ Crank-Nicolson scheme

**Number of forms per pad may vary slightly

APPENDIX V

TAPGEN Program Listings

1. Program BLOT Listing

```
C       PROGRAM BL3D (INPUT,OUTPUT,TAPE60=INPUT,TAPE61=OUTPUT)                 BL000010
C                                                                              BL000020
CCC        BL3D IS THE MAIN PROGRAM AND CALLS SEVERAL OTHER SUBROUTINES        BL000030
C                                                                              BL000040
        IMPLICIT REAL*8 (A-H,O-Z)                                              BL000050
        COMMON /C01/   A              ,  EPS           ,  JMAX        ,  JMAX1      , BL000060
       1               JMAX2     ,  KMAX          ,  KMAX1,JK       ,  PI        ,BL000070
       2               PSIMAX    ;  SO            ;  S1          ,  UINF      ,  BL000080
       3               WINF      ;  XO            ;  YO                          BL000090
        COMMON /C08/   IN        ,  IOUT                                         BL000100
        CALL ERRSET(207,256,10,0,0,209)                                        BL000110
        PI = 3.1415926536                                                      BL000120
        IN = 5                                                                 BL000130
        IOUT = 6                                                               BL000140
10      CONTINUE                                                               BL000150
        CALL INPUT                                                             BL000160
        CALL PRECAL                                                            BL000170
        CALL GEOM                                                              BL000180
        GO TO 10                                                               BL000190
        END                                                                    BL000200


        SUBROUTINE CALCUV (K,CS,SN,U,IMET,V)                                   BL000210
C                                                                              BL000220
CCC          CALCUV COMPUTES EITHER R OR X, DEPENDING UPON WHICH IS THE UNKBL000230
CCC          ** ARGUMENTS **                                                   BL000240
CCC                K - SUBSCRIPT IN PHI-DIRECTION.                             BL000250
CCC               CS - COS OF ANGLE UPON WHICH R OR X DEPENDS.                 BL000260
CCC               SN - SIN OF ANGLE UPON WHICH R OR X DEPENDS.                 BL000270
CCC                U - THE UNKNOWN - R OR X.                                   BL000280
CCC             IMET - = 2, IMPLIES SOLVING FOR X.                             BL000290
CCC                    = 1, IMPLIES SOLVING FOR R.                             BL000300
CCC                V - THE KNOWN VARIABLE - X OR R.                            BL000310
C                                                                              BL000320
        IMPLICIT REAL*8 (A-H,O-Z)                                              BL000330
        COMMON/CO30/   IA, JK2    ,  OTA          ,  PRX         ,  PRXOR      , BL000340
       1               XPOVA                                                   BL000350
        DIMENSION V(100)                                                       BL000360
        IF (IMET .NE. 1) GO TO 20                                              BL000370
10      CONTINUE                                                               BL000380
        XK = V(K)                                                              BL000390
        CALL CXPOVA (XK)                                                       BL000400
        U = RELLIP (CS,SN,K)                                                   BL000410
        GO TO 60                                                               BL000420
20      CONTINUE                                                               BL000430
        GO TO (30,40), IA                                                      BL000440
30      CONTINUE                                                               BL000450
        U = ELLSP(CS,SN,K)                                                     BL000460
```

```
      GO TO 50                                                          BL000470
40    CONTINUE                                                          BL000480
      U = ELLPD (CS,SN,K)                                               BL000490
50    CONTINUE                                                          BL000500
      CALL CXPOVA(U)                                                    BL000510
60    CONTINUE                                                          BL000520
      RETURN                                                            BL000530
      END                                                               BL000540


      SUBROUTINE COORD(V,U,UV,VOLD,UOLD,IMET,K,DS)                      BL000550
C                                                                       BL000560
CCC   SUBROUTINE COORD CALCULATES THE R AND X CYLINDRICAL COORDINATES   BL000570
CCC       OF A POINT ON THE SURFACE OF THE BODY.                        BL000580
CCC    ** ARGUMENTS **                                                  BL000590
CCC            V - R, IF IMET = 2.                                      BL000600
CCC                X, IF IMET = 1.                                      BL000610
CCC            U - X, IF IMET = 2.                                      BL000620
CCC                R, IF IMET = 1.                                      BL000630
CCC           UV - A PARTIAL DERIVATIVE - EITHER X W.R.T. R OR R        BL000640
CCC                     W.R.T. X.                                       BL000650
CCC         VOLD - EITHER R  OR  X FROM THE PREVIOUS STEP.              BL000660
CCC         UOLD - EITHER R  OR  X FROM THE PREVIOUS STEP.              BL000670
CCC         IMET - = 2, IMPLIES SOLVING EXPLICITLY FOR X, IMPLICITLY    BL000680
CCC                     FOR R.                                          BL000690
CCC                = 1, IMPLIES SOLVING EXPLICITLY FOR R, IMPLICITLY    BL000700
CCC                     FOR X.                                          BL000710
CCC            K - SUBSCRIPT IN THE PHI-DIRECTION.                      BL000720
CCC           DS - INCREMENTAL DISTANCE ALONG SURFACE OF THE BODY.      BL000730
C                                                                       BL000740
      IMPLICIT REAL*8 (A-H,O-Z)                                         BL000750
      COMMON /C01/  A           ,  EPS       ,  JMAX      ,  JMAX1     , BL000760
     1             JMAX2       , KMAX        , KMAX1,JK   ,  PI        ,BL000770
     2             PSIMAX      ; SO          ; SI         , UINF       , BL000780
     3             WINF        , XO          , YO                       BL000790
      COMMON /C04/  ALP         ,  COSPH(100),  DPHI(100) ,  DPSI      , BL000800
     1             DW(100)     , PHI(100)    , PSI        , PSIO       , BL000810
     2             SINPH(100),  SINTHB                                  BL000820
      COMMON /C05/  HPSI(100) ,  HW(100)     ,  HWO(100)                BL000830
      COMMON/C010/  LITER,JK1 ,  PI2         ;  SNTHB2                  BL000840
      COMMON/C023/  AO          ,  BO        ;  CO        ,  P         , BL000850
     1             RB          , RC                                     BL000860
      COMMON/C030/  IA, JK2     ,  OTA       ,  PRX       ,  PRXOR     , BL000870
     1             XPOVA                                                BL000880
      COMMON /EXT/ IBEXT                                                BL000890
      DIMENSION U(100), V(100), UV(100), UOLD(100), VOLD(100)           BL000900
      DIMENSION MET(2)                                                  BL000910
      IF (K.NE.0) GO TO 60                                              BL000920
      DS = DPSI                                                          BL000930
      DV = DS / DSQRT(1.0D0+UV(1)*UV(1))                                 BL000940
      V(1) = VOLD(1) + DV                                                BL000950
10    CONTINUE                                                           BL000960
      CALL CALCUV (1,COSPH(1),SINPH(1),US,IMET,V)                        BL000970
      U(1) = US                                                          BL000980
      UV(1) = (U(1) - UOLD(1)) / DV                                      BL000990
      DV = DS / DSQRT(1.0D0 + UV(1)*UV(1))                               BL001000
      VORG = V(1)                                                        BL001010
      V(1) = VOLD(1) + DV                                                BL001020
      IF (V(1).EQ.VORG) GO TO 20                                         BL001030
      IF (DABS(V(1)-VORG).GT.DABS(V(1)*EPS)) GO TO 10                    BL001040
20    CONTINUE                                                           BL001050
      CALL CALCUV (1,COSPH(1),SINPH(1),US,IMET,V)                        BL001060
      U(1) = US                                                          BL001070
      GO TO (30,40), IMET                                                BL001080
30    CONTINUE                                                           BL001090
      IF(IBEXT.EQ.0) GO TO 31                                            BL001100
      CALL DPRX(COSPH(1),U(1),PRX)                                       BL001110
      GO TO 32                                                           BL001120
   31 PRX = RB*(1.0D0-XPOVA) / (U(1)*OTA+YO*COSPH(1))                    BL001130
   32 PRXOR = (YO+U(1)*COSPH(1)*(1.0D0-RB/RC))*SINPH(1)/(U(1)*OTA +      BL001140
     1  YO*COSPH(1) + RB*PRX*(1.0D0-XPOVA))                              BL001150
      GO TO 50                                                           BL001160
40    CONTINUE                                                           BL001170
      IF(IBEXT.EQ.0) GO TO 41                                            BL001180
      CALL DPRX(COSPH(1),V(1),PRX)                                       BL001190
      GO TO 42                                                           BL001200
   41 PRX = RB*(1.0D0-XPOVA)/(V(1)*OTA + YO*COSPH(1))                    BL001210
   42 PRXOR = (YO+V(1)*COSPH(1)*(1.0D0-RB/RC))*SINPH(1)/(V(1)*OTA        BL001220
     1  +YO*COSPH(1) + RB*PRX*(1.0D0-XPOVA))                             BL001230
50    CONTINUE                                                           BL001240
      HPSI(1) = 1.0D0                                                    BL001250
      DV = 0.0D0                                                         BL001260
      MET(1) = 0                                                         BL001270
      MET(2) = 0                                                         BL001280
      RETURN                                                             BL001290
C           GENERAL CASE, K (NOT)=0.                                     BL001300
60    CONTINUE                                                           BL001310
      V(K+1) = V(K) + DV                                                 BL001320
      MET(IMET) = MET(IMET) + 1                                          BL001330
      CSPH3 = DCOS(PHI(K+1) + 0.5D0*DPHI(K))                             BL001340
      CSPH4 = DCOS(PHI(K+1) - 0.5D0*DPHI(K))                             BL001350
      SINPH3 = DSIN(PHI(K+1) + 0.5D0*DPHI(K))                            BL001360
      SINPH4 = DSIN(PHI(K+1) - 0.5D0*DPHI(K))                            BL001370
      IF ((MET(1).NE.1).AND.(MET(2).NE.1)) GO TO 110                     BL001380
70    CONTINUE                                                           BL001390
      CSPH1 = DCOS(PHI(K) + 0.5D0*DPHI(K))                               BL001400
      CSPH2 = DCOS(PHI(K) - 0.5D0*DPHI(K))                               BL001410
      SINPH1 = DSIN(PHI(K) + 0.5D0*DPHI(K))                              BL001420
      SINPH2 = DSIN(PHI(K) - 0.5D0*DPHI(K))                              BL001430
      CALL CALCUV(K,CSPH1,SINPH1,US,IMET,V)                                BL001440
      U1 = US                                                              BL001450
      CALL CALCUV (K,CSPH2,SINPH2,US,IMET,V)                               BL001460
      U2 = US                                                              BL001470
      GO TO (80,90), IMET                                                  BL001480
80    CONTINUE                                                             BL001490
      IF(IBEXT.EQ.0) GO TO 81                                              BL001500
      CALL DPRX(COSPH(K),U(K),PRX)                                         BL001510
      UV(K)=PRX                                                            BL001520
      GO TO 100                                                            BL001530
   81 UV(K) = RB * (1.0D0-XPOVA) / (U(K)*OTA+YO*COSPH(K))                  BL001540
      GO TO 100                                                            BL001550
90    CONTINUE                                                             BL001560
      IF(IBEXT.EQ.0) GO TO 91                                              BL001570
      CALL DPRX(COSPH(K),V(K),PRX)                                         BL001580
      UV(K)=PRX                                                            BL001590
      GO TO 100                                                            BL001600
   91 UV(K) = (V(K)*OTA+YO*COSPH(K)) / (RB*(1.0D0-XPOVA))                  BL001610
100   CONTINUE                                                             BL001620
      TERM1 = (U1 - U2) / DPHI(K)                                          BL001630
      GO TO 120                                                            BL001640
110   CONTINUE                                                             BL001650
      TERM1 = TERM2                                                        BL001660
120   CONTINUE                                                             BL001670
130   CONTINUE                                                             BL001680
      CALL CALCUV (K+1,CSPH3,SINPH3,US,IMET,V)                             BL001690
      U3 = US                                                              BL001700
      CALL CALCUV(K+1,CSPH4,SINPH4,US,IMET,V)                              BL001710
      U4 = US                                                              BL001720
      CALL CALCUV (K+1,COSPH(K+1),SINPH(K+1),US,IMET,V)                    BL001730
      U(K+1) = US                                                          BL001740
      GO TO (140,150), IMET                                                BL001750
140   CONTINUE                                                             BL001760
      IF(IBEXT.EQ.0) GO TO 141                                             BL001770
      CALL DPRX(COSPH(K+1),U(K+1),PRX)                                     BL001780
      UV(K+1)=PRX                                                          BL001790
      GO TO 160                                                            BL001800
  141 UV(K+1) = RB*(1.-XPOVA)/(U(K+1)*OTA+YO*COSPH(K+1))                   BL001810
      GO TO 160                                                            BL001820
150   CONTINUE                                                             BL001830
      IF(IBEXT.EQ.0) GO TO 151                                             BL001840
      CALL DPRX(COSPH(K+1),V(K+1),PRX)                                     BL001850
      UV(K+1)=PRX                                                          BL001860
      GO TO 160                                                            BL001870
  151 UV(K+1) = (V(K+1)*OTA + YO*COSPH(K+1)) / (RB*(1.0D0-XPOVA))          BL001880
160   CONTINUE                                                             BL001890
      DUDV = 0.5D0*(UV(K) + UV(K+1))                                       BL001900
      TERM2 = (U3 - U4) / DPHI(K)                                          BL001910
      DUDPH = 0.5D0 * (TERM1 + TERM2)                                      BL001920
      DV = -PI2*DUDPH*DUDV    / (1.0D0+DUDV   *DUDV   )*DW(K)              BL001930
      VORG = V(K+1)                                                        BL001940
      V(K+1) = V(K) + DV                                                BL001950
      IF (V(K+1).EQ.VORG) GO TO 170                                     BL001960
      IF (DABS(V(K+1)-VORG).GT.DABS(V(K+1)*EPS)) GO TO 130              BL001970
170   CONTINUE                                                          BL001980
      HWO(K) = HW(K)                                                    BL001990
      IF (IMET.NE.1) GO TO 190                                          BL002000
180   CONTINUE                                                          BL002010
      HW(K) = DSQRT((U(K+1)-U(K))**2 + (PI*(U(K+1)+U(K))*DW(K))**2      BL002020
     1     + (V(K+1) - V(K))**2) / DW(K)                                BL002030
      PRX = UV(K+1)                                                     BL002040
      PRXOR=(YO+U(K+1)*COSPH(K+1)*(1.0D0-RB/RC))*SINPH(K+1)/(U(K+1)*OTA BL002050
     1  +YO*COSPH(K+1)+RB*PRX*(1.0D0-XPOVA))                            BL002060
      GO TO 200                                                         BL002070
190   CONTINUE                                                          BL002080
      HW(K) = DSQRT((U(K+1)-U(K))**2 + (PI*(V(K+1)+V(K))*DW(K))**2      BL002090
     1     + (V(K+1) - V(K))**2) / DW(K)                                BL002100
      PRX = 1.0D0/UV(K+1)                                               BL002110
     1  +YO*COSPH(K+1)+RB*PRX*(1.0D0-XPOVA)                             BL002120
200   CONTINUE                                                          BL002130
      DELV = V(K+1) - VOLD(K+1)                                         BL002140
      UVH = (U(K+1) - UOLD(K+1)) / DELV                                 BL002150
      DS = DSQRT(1.0D0 + UVH*UVH) * DELV                                BL002160
      HPSI(K+1) = DS/DPSI                                               BL002170
      RETURN                                                            BL002180
      END                                                               BL002190


      SUBROUTINE CXPOVA (X)                                             BL002200
C                                                                       BL002210
CCC       CXPOVA CALCULATES THE RELATION XPRIME/A.                      BL002220
CCC       **ARGUMENTS **                                                BL002230
CCC           X - X-COORDINATE.                                         BL002240
C                                                                       BL002250
      IMPLICIT REAL*8 (A-H,O-Z)                                         BL002260
      COMMON /CO1/   A          , EPS          , JMAX         , JMAX1        , BL002270
     1          JMAX2       , KMAX         , KMAX1,JK     , PI          ,BL002280
     2          PSIMAX      , SO           , S1           , UINF         , BL002290
     3          WINF        , XO           , YO                            BL002300
      COMMON/CO30/   IA, JK2    , OTA          , PRX          , PRXOR        , BL002310
     1          XPOVA                                                   BL002320
      GO TO (10,20), IA                                                 BL002330
10    CONTINUE                                                          BL002340
      XPOVA = (X + XO) /A                                               BL002350
      GO TO 30                                                          BL002360
20    CONTINUE                                                          BL002370
      XPOVA = 0.0D0                                                     BL002380
30    CONTINUE                                                          BL002390
      RETURN                                                            BL002400
      END                                                               BL002410


      DOUBLE PRECISION FUNCTION ELLPD (COSP,SINP,K)                     BL002420
C                                                                       BL002430
CCC      ELLPD DEFINES X WHEN THE BODY GEOMETRY IS AN ELLIPTIC          BL002440
CCC    PARABOLOID.                                                      BL002450
CCC      ** ARGUMENTS **                                                BL002460
CCC           COSP - COS OF THE ANGLE.                                  BL002470
CCC           SINP - SIN OF THE ANGLE.                                  BL002480
CCC              K - SUBSCRIPT IN THE PHI-DIRECTION.                    BL002490
C                                                                       BL002500
      IMPLICIT REAL*8 (A-H,O-Z)                                         BL002510
      COMMON /C01/   A            , EPS         , JMAX        , JMAX1     , BL002520
     1               JMAX2     ,  KMAX       ,  KMAX1,JK     ,  PI      ,BL002530
     2               PSIMAX    ,  SO         ;  S1         ,  UINF     ,  BL002540
     3               WINF      ,  XO         ,  YO                        BL002550
      COMMON /C02/    ALPHA(100),  DN(50)     ,  DN2(50)    ,  R(100)    , BL002560
     1               RWRTX(100),  S(100)     ,  W(100)     ,  X(100)      BL002570
      COMMON/C023/    AO          , BO          , CO          , P         , BL002580
     1               RB        ,  RC                                      BL002590
      COMMON/C030/    IA, JK2     ,  OTA         , PRX         , PRXOR     , BL002600
     1               XPOVA                                                BL002610
      OTA = COSP*COSP + RB/RC*SINP*SINP                                 BL002620
      RP2 = R(K)*R(K) + 2.0D0*YO * R(K)*COSP + YO*YO                    BL002630
      D = RB*RP2/(R(K)*R(K)*OTA + 2.0D0*YO*R(K)*COSP+YO*YO)             BL002640
      ELLPD = 0.5D0*RP2/D - XO                                          BL002650
      RETURN                                                            BL002660
      END                                                               BL002670


      DOUBLE PRECISION FUNCTION ELLSP(COSP,SINP,K)                      BL002680
C                                                                       BL002690
CCC       THIS FUNCTION DEFINES X WHEN THE BODY GEOMERTY IS AN ELLIPSOID.BL002700
CCC      ** ARGUMENTS **                                                BL002710
CCC           COSP - COS OF THE ANGLE.                                  BL002720
CCC           SINP - SIN OF THE ANGLE.                                  BL002730
CCC              K - SUBSCRIPT IN THE PHI DIRECTION.                    BL002740
C                                                                       BL002750
      IMPLICIT REAL*8 (A-H,O-Z)                                         BL002760
      COMMON /C01/   A            , EPS         , JMAX        , JMAX1     , BL002770
     1               JMAX2     ,  KMAX       ,  KMAX1,JK     ,  PI      ,BL002780
     2               PSIMAX    ,  SO         ;  S1         ,  UINF     ,  BL002790
     3               WINF      ,  XO         ,  YO                        BL002800
      COMMON /C02/    ALPHA(100),  DN(50)     ,  DN2(50)    ,  R(100)    , BL002810
```

```
DATE  04/20/77

     1              RWRTX(100),  S(100)     ,  W(100)     ,  X(100)        BL002820
      COMMON/C023/    AO        ,  BO        ,  CO         ,  P          ,  BL002830
     1              RB         ,  RC                                       BL002840
      COMMON/C030/    IA, JK2   ,  OTA       ,  PRX        ,  PRXOR      ,  BL002850
     1              XPOVA                                                  BL002860
      COMMON /EXT/ IBEXT                                                   BL002870
      DIMENSION ZSLO(2)                                                    BL002880
      OTA = COSP*COSP + RB/RC*SINP*SINP                                    BL002890
      IF(IBEXT.EQ.1) GO TO 10                                              BL002900
      RP2 = R(K)*R(K) + 2.0D0*YO * R(K)*COSP + YO*YO                       BL002910
      D = RB *RP2/(R(K)*R(K)*OTA + 2.0D0*YO*R(K)*COSP + YO*YO)             BL002920
      ELLSP = A - XO - DSQRT(A*A-A*RP2/D)                                  BL002930
      GO TO 20                                                             BL002940
   10 TANP=SINP/COSP                                                       BL002950
      PH1=DATAN(TANP)                                                      BL002960
      PP=PH1                                                               BL002970
      ZSLO(1)=R(K)                                                         BL002980
      CALL SLOALL(PP,ZSLO,1,13,2,0,0,3,0)                                  BL002990
      ELLSP=ZSLO(2)                                                        BL003000
   20 RETURN                                                               BL003010
      END                                                                  BL003020


      SUBROUTINE INPUT                                                     BL003030
C                                                                          BL003040
C        INPUT READS INPUT DATA AND INITIALIZES A FEW PARAMETERS.          BL003050
C                                                                          BL003060
      IMPLICIT REAL*8 (A-H,O-Z)                                            BL003070
      COMMON /C01/    A         ,  EPS       ,  JMAX       ,  JMAX1      ,  BL003080
     1              JMAX2      ,  KMAX      ,  KMAX1,JK    ,  PI         ,BL003090
     2              PSIMAX     ,  SO        ,  S1         ,  UINF       ,  BL003100
     3              WINF       ,  XO        ,  YO                          BL003110
      COMMON /C02/    ALPHA(100),  DN(50)    ,  DN2(50)    ,  R(100)     ,  BL003120
     1              RWRTX(100),  S(100)     ,  W(100)     ,  X(100)        BL003130
      COMMON /C04/    ALP       ,  COSPH(100),  DPHI(100)  ,  DPSI       ,  BL003140
     1              DW(100)    ,  PHI(100)  ,  PSI        ,  PSIO       ,  BL003150
     2              SINPH(100),  SINTHB                                    BL003160
      COMMON /C08/    IN        ,  IOUT                                    BL003170
      COMMON/C010/    LITER,JK1 ,  PI2       ,  SNTHB2                     BL003180
      COMMON/C017/    IPRT      ,  KPRT                                    BL003190
      COMMON/C023/    AO        ,  BO        ,  CO         ,  P          ,  BL003200
     1              RB         ,  RC                                       BL003210
      COMMON/C025/    CAPTH     ,  CPTHM     ,  CPTHP      ,  CPTH2        BL003220
      COMMON/C041/    CRI       ,  KHALF                                   BL003230
      COMMON /EXT/ IBEXT                                                   BL003240
      DIMENSION TITLE(9)                                                   BL003250
      READ (4,9000,END=10) (TITLE(I),I=1,9)                                BL003260
      GO TO 20                                                             BL003270
```

```
C     IF (EOF,60) 10,20                                                 BL003280
10    CONTINUE                                                          BL003290
      WRITE(6,11) PSI,PSIMAX                                            BL003300
   11 FORMAT(///10X,'PSI =',F10.4,5X,'GREATER THAN  PSIMAX=',F10.4,10X, BL003310
     1'SO      THE END')                                                BL003320
      STOP                                                              BL003330
20    CONTINUE                                                          BL003340
      WRITE(IOUT,9900) (TITLE(I),I=1,9)                                 BL003350
      READ (4,9010) JMAX, KMAX, LITER, IPRT, KPRT, IBEXT,  PSIMAX, UINF BL003360
      KMAX1 = KMAX - 1                                                  BL003370
      JMAX1 = JMAX - 1                                                  BL003380
      READ (4,9020) (DW(K),K=1,KMAX1)                                   BL003390
      READ (4,9020) (DN(J),J=1,JMAX1)                                   BL003400
      READ (4,9020) DPSI                                                BL003410
      READ (4,9020) A,RB,RC,EPS,ALP,CAPTH,CRI                           BL003420
      KHALF = KMAX/2 + 1                                                BL003430
      IF (MOD(KMAX,2) .NE. 0) KHALF = KHALF + 1                         BL003440
C                                                                       BL003450
C                                                                       BL003460
C                                                                       BL003470
      RETURN                                                            BL003480
C                                                                       BL003490
 9000 FORMAT (5X,9A8)                                                   BL003500
 9900 FORMAT (1H1,20X,9A8)                                              BL003510
 9010 FORMAT (5X,6I5,3E10.0)                                            BL003520
 9020 FORMAT (5X,7E10.0)                                                BL003530
      END                                                               BL003540
      SUBROUTINE PARVAL (K,IP)                                          BL003550
C                                                                       BL003560
CCC     PARVAL - PARAMETER EVALUATION   - -  KEY PARAMETERS NEEDED FOR  BL003570
CCC   THE SOLUTION OF F,G, ADD V ARE CALCULATED IN THIS SUBROUTINE.     BL003580
CCC        ** ARGUMENTS **                                              BL003590
CCC             K - SUBSCRIPT IN THE PHI-DIRECTION.                     BL003600
CCC            IP - SUBSCRIPT IN THE PSI-DIRECTION.                     BL003610
C                                                                       BL003620
      IMPLICIT REAL*8 (A-H,O-Z)                                         BL003630
       REAL*8 KAPPAS, KAPPAW                                            BL003640
       COMMON /C01/    A          ,  EPS        ,  JMAX       ,  JMAX1     , BL003650
     1               JMAX2      ,  KMAX       ,  KMAX1,JK     ,  PI        ,BL003660
     2               PSIMAX     ,  SO         ,  S1         ,  UINF       , BL003670
     3               WINF       ,  XO         ,  YO                         BL003680
       COMMON /C02/    ALPHA(100),  DN(50)     ,  DN2(50)    ,  R(100)    , BL003690
     1               RWRTX(100),  S(100)     ,  W(100)     ,  X(100)        BL003700
       COMMON /C03/    F(50,100) ,  FO(50,100),  G(50,100) ,  GO(50,100), BL003710
     1               UE(100)    ,  UEO(100)   ,  V(50,100) ,  WE(100)    ,  BL003720
     2               WEO(100)                                           BL003730
       COMMON /C04/    ALP        ,  COSPH(100),  DPHI(100) ,  DPSI      , BL003740
     1               DW(100)    ,  PHI(100)   ,  PSI        ,  PSIO       , BL003750
     2               SINPH(100),  SINTHB                                BL003760
       COMMON /C05/    HPSI(100) ,  HW(100)    ,  HWO(100)                 BL003770
       COMMON /C06/    ALBET2     ,  ALBET3     ,  ALP2       ,  BETA1     , BL003780
     1              BETA2      ,  BETA3      ,  BETA4     ,  KAPPAS     ,  BL003790
     2              KAPPAW                                                  BL003800
      COMMON /C07/    ALPHO(50) ,  ALPH1(50) ,  ALPH2(50) ,  ALPH3(50) ,  BL003810
     1              ALPH4(50) ,  ALPH5(50) ,  ALPH6(50)                      BL003820
      COMMON /C09/    TOL                                                   BL003830
      COMMON/C010/    LITER,JK1 ,  PI2        ,  SNTHB2                      BL003840
      COMMON/C011/    PIH                                                   BL003850
      COMMON/C012/    H         ,  HH         ,  WE1                         BL003860
      COMMON/C013/    COSAL     ;  SINAL                                    BL003870
      COMMON/C014/    COSPH2    ,  DW2        ,  PRXS       ,  PRXORS     ,  BL003880
     1              R2        ,  SINPH2     ,  XPOVAS     ,  X2             BL003890
      COMMON/C015/    ROLD(100) ,  SOLD(100) ,  THB(100)   ,  XOLD(100) ,  BL003900
     1              XWRTR(100)                                              BL003910
      COMMON/C018/    CAPW(100)                                             BL003920
      COMMON/C019/    PWOU(100) ,  PUOW                                     BL003930
      COMMON/C020/    ALBAV     ,  ALPAV                                    BL003940
      COMMON/C021/    ALB1(100) ,  ALB2       ,  ALB3       ,  BETA5      ,  BL003950
     1              BETA6                                                   BL003960
      COMMON/C023/    AO        ,  BO         ,  CO         ,  P          ,  BL003970
     1              RB        ,  RC                                         BL003980
      COMMON/C024/    GPR       ,  PHIPR      ,  RP         ,  UPHP       ,  BL003990
     1              USP       ,  XP                                         BL004000
      COMMON/C025/    CAPTH     ,  CPTHM      ,  CPTHP      ,  CPTH2         BL004010
      COMMON/C026/    XIKA      ,  AL3T                                     BL004020
      COMMON/C029/     WE1AV                                                BL004030
      COMMON/C030/    IA, JK2   ,  OTA        ,  PRX        ,  PRXOR      ,  BL004040
     1              XPOVA                                                   BL004050
      COMMON/C031/    UXP       ,  UYP        ,  UZP                         BL004060
      COMMON/C032/    CU        ;  CV         ;  CW                          BL004070
      COMMON/C036/    ICAP                                                  BL004080
      COMMON/C038/    HPS       ,  FKWPSI                                   BL004090
      COMMON/C040/    ALSTG                                                 BL004100
      COMMON/C042/    UE2       ,  WE2                                      BL004110
      COMMON/C043/    DRDSI     ;  XTST                                     BL004120
      COMMON /C050/ KDUM                                                    BL004130
      KDUM=K                                                                BL004140
      IF (IP .EQ. 1) GO TO 100                                              BL004150
      UEO(K+1) = UE(K+1)                                                    BL004160
      WEO(K+1) = WE(K+1)                                                    BL004170
      Y = R(K+1)*COSPH(K+1)                                                 BL004180
      Z = R(K+1)*SINPH(K+1)                                                 BL004190
      CALL UXYZ (Y,Z,XPOVAS)                                                BL004200
      UE(K+1) = (UXP + (UYP*COSPH(K+1)+UZP*SINPH(K+1))*PRXS)                BL004210
     1   /DSQRT(1.+PRXS*PRXS)                                               BL004220
C                                                                           BL004230
C        THE CALCULATION OF UE INVOLVES SETTING UE TO A RADICAL WHICH       BL004240
C     CAN ASSUME EITHER A (+) OR (-) SIGN.  A TEST IS MADE CONCERNING       BL004250
C     THE SIGNS OF TWO PARTIAL DERIVATIVES IN ORDER TO DETERMINE THE        BL004260
C     APPROPRIATE SIGN FOR UE.                                              BL004270
C        IT IS ASSUMED THAT FOR X GREATER THAN OR = XTST THE                BL004280
C     (+) SIGN SHOULD BE USED.                                              BL004290
```

DATE 04/20/77

```
C                                                                       BL004300
      IF (X(K+1).GE.XTST) GO TO 10                                      BL004310
      IF ((PRXS*DRDSI) .LT. 0.) UE(K+1) = - UE(K+1)                     BL004320
10    CONTINUE                                                          BL004330
      WE(K+1) = ((UYP*COSPH(K+1)+UZP*SINPH(K+1)-UXP*PRXS)*PRXORS        BL004340
     1  + UZP*COSPH(K+1) - UYP*SINPH(K+1))/DSQRT(1.+(1.+PRXS*PRXS)      BL004350
     1    * PRXORS*PRXORS)                                              BL004360
      IF (K.EQ.KMAX1) GO TO 80                                          BL004370
      Y = R2*COSPH2                                                     BL004380
      Z = R2*SINPH2                                                     BL004390
      CALL UXYZ (Y,Z,XPOVA)                                             BL004400
      UE2 = (UXP + (UYP*COSPH2+UZP*SINPH2)*PRX)/DSQRT(1.+PRX*PRX)       BL004410
      IF (X2.GE.XTST) GO TO 20                                          BL004420
      DRDSIN = (R(K+2) - ROLD(K+2)) / DPSI                              BL004430
      IF ((PRX*DRDSIN) .LT. 0.) UE2 = -UE2                              BL004440
20    CONTINUE                                                          BL004450
      WE2 = ((UYP*COSPH2+UZP*SINPH2-UXP*PRX )*PRXOR  + UZP*COSPH2-UYP   BL004460
     1  *SINPH2)/DSQRT(1.+(1.+PRX *PRX )*PRXOR *PRXOR )                 BL004470
      IF (K.NE.0) GO TO 80                                              BL004480
30    CONTINUE                                                          BL004490
      CAPW(K+1) = WE(K+1)                                               BL004500
      H2 = (HPSI(K+2) - HPSI(K+1)) / (DW(K+1) * DW(K+1))                BL004510
      WE10 = WE1                                                        BL004520
      WE1 = WE2/(PI2*DW2)                                               BL004530
      WE1AV = .5 * (WE1 + WE10)                                         BL004540
      USUBE = .5 * (UEO(K+1) + UE(K+1))                                 BL004550
      RB1 = .5 * (ROLD(K+1) + R(K+1))                                   BL004560
      KAPPAW = (R(K+1) - ROLD(K+1)) / (RB1*DPSI)                        BL004570
      FKPSO = FXPS                                                      BL004580
      FKPS = KAPPAW * PSI                                               BL004590
      DWE1 = (WE1 - WE10) / DPSI                                        BL004600
      HPS = .5*(PSI + PSIO)                                             BL004610
      FKWPSI = KAPPAW * HPS                                             BL004620
      IF (WE1AV.LE..0000000001) GO TO 40                                BL004630
      BETA4 = (HPS*DWE1) / WE1AV                                        BL004640
      KAPPAS = H2 * USUBE / (PI2*PI*RB1*WE1AV)                          BL004650
      GO TO 50                                                          BL004660
40    CONTINUE                                                          BL004670
      BETA4 = 0.0                                                       BL004680
      KAPPAS = FKWPSI / HPS                                             BL004690
50    CONTINUE                                                          BL004700
      ALBET3 = HPS * WE1AV / (RB1*USUBE)                                BL004710
      IF (IP.EQ.2) GO TO 60                                             BL004720
      BETA1 = (UE(K+1) - UEO(K+1))/(DPSI*.5*(UE(K+1)/PSI+UEO(K+1)/PSIO))BL004730
      GO TO 70                                                          BL004740
60    CONTINUE                                                          BL004750
      BETA1 = (UE(K+1) - UEO(K+1)) / (DPSI*.5*(UE(K+1)/PSI + UO1))      BL004760
70    CONTINUE                                                          BL004770
      BETA6 = BETA4                                                     BL004780
      BETA2 = 0.0                                                       BL004790
      ALSTG = ALPHA(K+1)                                                BL004800
      ALPHA(K+1) = 0.0                                                  BL004810
      ALB1(K+1) = ALPHA(K+1)                                            BL004820
      ALB2 = ALPHA(K+1)                                                 BL004830
      ALB3 = 1.0                                                        BL004840
      ALBET2 = 0.0                                                      BL004850
      XIKA = 0.0                                                        BL004860
      BETA3 = 0.0                                                       BL004870
      GO TO 90                                                          BL004880
 80   CONTINUE                                                          BL004890
      CAPW(K+1) = UE(K+1)                                               BL004900
      CALL AVERG(K,IP)                                                  BL004910
      IF(ALB1(K+1) .LT. 1.0D-06) GO TO 90                               BL004920
      XIKA = HPS * KAPPAS / ALB1(K+1)                                   BL004930
 90   CONTINUE                                                          BL004940
      ALP2 = ALPHA(K+1) * ALPHA(K+1)                                    BL004950
      GO TO 210                                                         BL004960
 100  CONTINUE                                                          BL004970
      IF (K.NE.0) GO TO 110                                             BL0C4980
      CSP = COSPH(1)                                                    BL004990
      SNP = SINPH(1)                                                    BL005000
      GO TO 120                                                         BL005010
 110  CONTINUE                                                          BL005020
      PHIAV = .5*(PHI(K) + PHI(K+1))                                    BL005030
      CSP = DCOS(PHIAV)                                                 BL005040
      SNP = DSIN(PHIAV)                                                 BL005050
 120  CONTINUE                                                          BL005060
      COMDE = S0*CSP*CSP + S1*SNTHB2*SNP*SNP                            BL005070
      UE(K+1) = 0.0                                                     BL005080
      WE(K+1) = 0.0                                                     BL005090
      CAPW(K+1) = WE(K+1)                                               BL005100
      IF (K .EQ. 1) WE1 = WE(2) / (PI2*DW(2))                           BL005110
      ALPHA(K+1) = (S1-S0)*SINTHB*SNP*CSP/COMDE                         BL005120
      ALP2 = ALPHA(K+1) * ALPHA(K+1)                                    BL005130
      PWOU(K+1) = (S1-S0)*(CSP*CSP-SNTHB2*SNP*SNP)/COMDE                BL005140
      PUOW = 1.0                                                        BL005150
      FKPS = 1.0                                                        BL005160
      ALB3 = 1.0                                                        BL005170
      IF (DABS(S0-S1).GT..0000000001) GO TO 140                         BL005180
 130  CONTINUE                                                          BL005190
      XIKA = 2.0                                                        BL005200
      BETA4 = 2.0                                                       BL005210
      GO TO 150                                                         BL005220
 140  CONTINUE                                                          BL005230
      XIKA = 0.0                                                        BL005240
      BETA4 = 1.0                                                       BL005250
 150  CONTINUE                                                          BL005260
      IF (K.EQ.0) GO TO 170                                             BL005270
 160  CONTINUE                                                          BL005280
      HOLD = H                                                          BL005290
 170  CONTINUE                                                          BL005300
      H = SINTHB / (CSP*CSP + SNTHB2*SNP*SNP)                           BL005310
      IF (K.EQ.0) GO TO 190
180   CONTINUE
      ALBET3 = PWOU(K+1)
      GO TO 200
190   CONTINUE
      UO1 = H*SO/SINTHB
      WO2 = (S1-SO) * SINPH(2)*COSPH(2) *SINTHB/
     1  (COSPH(2)*COSPH(2)+SNTHB2*SINPH(2)*SINPH(2))
      ALBET3 = WO2/(PI2*H*UO1*DW(1))
200   CONTINUE
      PIH = H * PI2
      ALPAV = ALPHA(K+1)
      COMDE = SO*COSPH(K+1)*COSPH(K+1)+S1*SNTHB2*SINPH(K+1)*SINPH(K+1)
      ALPHA(K+1) = (S1-SO)*SINTHB*SINPH(K+1)*COSPH(K+1)/COMDE
210   CONTINUE
      RETURN
      END


      SUBROUTINE PRECAL
C
CCC   SUBROUTINE PRECAL DOES ONE-TIME CALCULATIONS AND INITIALIZES
CCC       VARIABLES.  ALSO THE INITIAL INPUT IS PRINTED OUT IN THIS
CCC       SUBROUTINE.
C
      IMPLICIT REAL*8 (A-H,O-Z)
      COMMON /CO1/    A          ,  EPS        ,  JMAX       ,  JMAX1      ,
     1              JMAX2      , KMAX        , KMAX1,JK    ,  PI         ,
     2              PSIMAX     , SO          , S1          , UINF        ,
     3              WINF       , XO          , YO
      COMMON /CO2/    ALPHA(100),  DN(50)      ,  DN2(50)    ,  R(100)     ,
     1              RWRTX(100),  S(100)      , W(100)      , X(100)
      COMMON /CO3/    F(50,100) ,  FO(50,100),  G(50,100)  ,  GO(50,100),
     1              UE(100)    , UEO(100)    , V(50,100)  , WE(100)     ,
     2              WEO(100)
      COMMON /CO4/    ALP        ,  COSPH(100),  DPHI(100)  ,  DPSI       ,
     1              DW(100)    , PHI(100)    , PSI         , PSIO        ,
     2              SINPH(100),  SINTHB
      COMMON /CO5/    HPSI(100) ,  HW(100)     ,  HWO(100)
      COMMON /CO6/    ALBET2     ,  ALBET3     ,  ALP2       ,  BETA1      ,
     1              BETA2      , BETA3       , BETA4       , KAPPAS      ,
     2              KAPPAW
      COMMON /CO8/    IN         ,  IOUT
      COMMON /CO9/    TOL
      COMMON/CO10/    LITER,JK1 ,  PI2         ,  SNTHB2
      COMMON/CO13/    COSAL      ,  SINAL
      COMMON/CO15/    ROLD(100) , SOLD(100) , THB(100) , XOLD(100) ,
     1              XWRTR(100)
```

```
      COMMON/C017/    IPRT       ,  KPRT                                  BL005780
      COMMON/C023/    AO         ,  BO         ,  CO         ,  P         , BL005790
     1               RB         ,  RC                                     BL005800
      COMMON/C025/    CAPTH      ,  CPTHM      ,  CPTHP      ,  CPTH2     BL005810
      COMMON/C030/    IA, JK2    ,  OTA        ,  PRX        ,  PRXOR     , BL005820
     1               XPOVA                                                BL005830
      COMMON/C032/    CU         ,  CV         ,  CW                      BL005840
      COMMON/C034/    ICON       ,  JTST       ,  NIT                     BL005850
      COMMON/C036/    ICAP                                                BL005860
      COMMON/C041/    CRI        ,  KHALF                                 BL005870
      COMMON/C043/    DRDSI      ,  XTST                                  BL005880
      COMMON /EXT/ IBEXT                                                  BL005890
      DIMENSION      DNSUM(50)                                            BL005900
      TOL = .001                                                          BL005910
      DNSUM(1) = 0.0                                                      BL005920
      ALP = .0174533 * ALP                                                BL005930
      PI2 = 2.*PI                                                         BL005940
      JMAX1 = JMAX - 1                                                    BL005950
      JMAX2 = JMAX - 2                                                    BL005960
      KMAX1 = KMAX - 1                                                    BL005970
      ICAP = CAPTH                                                        BL005980
      CPTHM = 1. - CAPTH                                                  BL005990
      CPTHP = 1. + CAPTH                                                  BL006000
      CPTH2 = 2. - CAPTH                                                  BL006010
      JTST = JMAX/2 + 1                                                   BL006020
C                                                                         BL006030
      XTST = .1                                                           BL006040
      RWRTX(1) = 0.0                                                      BL006050
      XOLD(1) = -DPSI                                                     BL006060
      ROLD(1) = 0.0                                                       BL006070
      S(1) = 0.0                                                          BL006080
      X(1) = -DPSI                                                        BL006090
      PSI     = -DPSI                                                     BL006100
      PSIO = PSI                                                          BL006110
      XWRTR(1) = 0.0                                                      BL006120
      DO 10 J=1,JMAX1                                                     BL006130
      DN2(J) = DN(J) * DN(J)                                              BL006140
   10 CONTINUE                                                            BL006150
      F(1,1) = 0.0                                                        BL006160
      G(1,1) = 0.0                                                        BL006170
      FGINCR = 1./JMAX1                                                   BL006180
      F(JMAX,1) = 1.0                                                     BL006190
      G(JMAX,1) = 1.0                                                     BL006200
      DO 20 J=2,JMAX1                                                     BL006210
      F(J,1) = F(J-1,1) + FGINCR*F(JMAX,1)                                BL006220
      G(J,1) = G(J-1,1) + FGINCR*G(JMAX,1)                                BL006230
   20 CONTINUE                                                            BL006240
      PHI(1) = 0.0                                                        BL006250
      DPHI(1) = PI2*DW(1)                                                 BL006260
      COSAL = DCOS(ALP)                                                   BL006270
      SINAL = DSIN(ALP)                                                   BL006280
      DO 30 K=2,KMAX1                                                   BL006290
      DPHI(K) = PI2 * DW(K)                                             BL006300
      PHI(K) = PHI(K-1) + DPHI(K-1)                                     BL006310
30    CONTINUE                                                          BL006320
      PHI(KMAX) = PHI(KMAX1) + DPHI(KMAX1)                              BL006330
      DO 40 K=1,KMAX                                                    BL006340
      SINPH(K) = DSIN(PHI(K))                                           BL006350
      COSPH(K) = DCOS(PHI(K))                                           BL006360
      HW(K) = 0.0                                                       BL006370
      SOLD(K) = 0.0                                                     BL006380
      R(K) = 0.0                                                        BL006390
      FO(1,K) = 0.0                                                     BL006400
      GO(1,K) = 0.0                                                     BL006410
      FO(JMAX,K) = 1.0                                                  BL006420
      GO(JMAX,K) = 1.0                                                  BL006430
40    CONTINUE                                                          BL006440
C**        COEFFICIENTS FOR SEVERAL GEOMETRIES ARE DEFINED.             BL006450
      IF (A.LT.1.E+06)   GO TO 60                                       BL006460
50    CONTINUE                                                          BL006470
      IA = 2                                                            BL006480
      GO TO 70                                                          BL006490
60    CONTINUE                                                          BL006500
      IA = 1                                                            BL006510
70    CONTINUE                                                          BL006520
      IF (IA.EQ.2)   GO TO 110                                          BL006530
80    CONTINUE                                                          BL006540
      IF (A.NE.RB) GO TO 100                                            BL006550
C*****   SPHERE                                                         BL006560
90    CONTINUE                                                          BL006570
      AO = 2. / 3.                                                      BL006580
      BO = AO                                                           BL006590
      CO = BO                                                           BL006600
      GO TO 130                                                         BL006610
C*****   ELLIPSIOD OF REVOLUTION                                        BL006620
100   CONTINUE                                                          BL006630
      E = DSQRT(1.-RB/A)                                                BL006640
      AO = 2.*(1.-E*E)/(E**3.)*(.5*DLOG((1.+E)/(1.-E)) - E)             BL006650
      BO = 1./(E*E) - (1.-E*E) / (2.*E**3.)*DLOG((1.+E)/(1.-E))         BL006660
      CO = BO                                                           BL006670
      GO TO 130                                                         BL006680
110   CONTINUE                                                          BL006690
      IF (RB.EQ.RC) GO TO 120                                           BL006700
C*****   ELLIPTIC PARABOLOID                                            BL006710
      Q = DSQRT(RC/RB)                                                  BL006720
      AO = 0.0                                                          BL006730
      BO = 2.*Q / (1. + Q)                                              BL006740
      CO = 2./(1. + Q)                                                  BL006750
      GO TO 130                                                         BL006760
C*****   PARABOLOID OF REVOLUTION                                       BL006770
120   CONTINUE                                                          BL006780
      AO = 0.0                                                          BL006790
      BO = 1.0                                                          BL006800
      CO = BO                                                           BL006810
 130  CONTINUE                                                          BL006820
      P = (2.-AO) / (2.-BO) * DTAN(ALP)                                 BL006830
      CU   = UINF * COSAL                                               BL006840
      CV   = -UINF * SINAL                                              BL006850
      CW = 0.0                                                          BL006860
      S1 = 2.*UINF*COSAL*DSQRT(1.+RB*P*P/A) / (RC*(2.-AO))              BL006870
      SO = S1 *(RC/RB) * (1.+RB*P*P/A) / (1.+P*P)                       BL006880
      GO TO (140,150), IA                                               BL006890
 140  CONTINUE                                                          BL006900
      SQ = DSQRT(1. + RB*P*P/A)                                         BL006910
      XO = A * (1. - 1./SQ)                                             BL006920
      YO = RB*P/SQ                                                      BL006930
      GO TO 160                                                         BL006940
 150  CONTINUE                                                          BL006950
      XO = RB * .5*P*P                                                  BL006960
      YO = RB*P                                                         BL006970
 160  CONTINUE                                                          BL006980
      SINTHB = 1. / DSQRT(1.+P*P)                                       BL006990
      SNTHB2 = SINTHB*SINTHB                                            BL007000
      PIH = PI2 * SINTHB                                                BL007010
      ALPHA(1) = 0.0                                                    BL007020
      ALBET2 = 0.0                                                      BL007030
      ALBET3 = -(1.-S1/SO)                                              BL007040
      ALP1 = ALP/.0174533                                               BL007050
C                                                                       BL007060
C**        PRINT OUT INPUT                                              BL007070
C                                                                       BL007080
      WRITE (IOUT,9910) JMAX,PSIMAX,A,XO,KMAX,DPSI,SO,YO,LITER,         BL007090
     1     EPS,S1,CKA,IPRT,UINF,CAPTH,CKC,KPRT,ALP1,RB,E,IBEXT,RC,CRI   BL007100
      WRITE(IOUT,9920)                                                  BL007110
      DO 170 K=1,KMAX1                                                  BL007120
      WRITE (IOUT,9930) K, DW(K)                                        BL007130
 170  CONTINUE                                                          BL007140
      WRITE(IOUT,9940)                                                  BL007150
      DO 180 J=1,JMAX1                                                  BL007160
      DNSUM(J+1) = DNSUM(J) + DN(J)                                     BL007170
      WRITE(IOUT,9950) J, DN(J), DNSUM(J+1)                             BL007180
 180  CONTINUE                                                          BL007190
      IF(IBEXT.EQ.0) GO TO 190                                          BL007200
      CALL BODYG                                                        BL007210
  190 RETURN                                                            BL007220
C                                                                       BL007230
 9910 FORMAT (1HO,14X'JMAX = ',I5,9X'PSIMAX = ',E11.4,9X'A = ',E11.4,8X,BL007240
     1                 'XO = ',E11.4/                                   BL007250
     2               15X'KMAX = ',I5,11X'DPSI = ',E11.4,8X'SO = ',E11.4,8X,BL007260
     3                 'YO = ',E11.4/                                   BL007270
     4            14X'LITER = ',I5,12X',EPS = ',E11.4,8X'S1 = ',E11.4,8X, BL007280
     5                 'KA = ',E11.4/                                   BL007290
     6               15X',IPRT = ',I5,11X'UINF = ',E11.4,5X'CAPTH = ',E11.4BL007300
```

DATE 04/20/77

```
      7            ,8X,'KC = ',E11.4/                                       BL007310
      8             15X'KPRT = ',I5,10X'ALPHA = ',E11.4,8X'RB = ',F11.4,9XBL007320
      9                ,',E = ',E11.4/                                      BL007330
      1             32X,'IBEXT = ',I5,19X,'RC = ',E11.4,7X'CRI = ',E11.4)  BL007340
 9920 FORMAT (1H0// 14X'DW')                                                BL007350
 9930 FORMAT (2X,I3,4X,E11.4)                                               BL007360
 9940 FORMAT (1H0//4X,'J',9X,'DN',9X,'ETA')                                 BL007370
 9950 FORMAT (2X,I3,4X,E11.4,4X,E11.4)                                      BL007380
      END                                                                   BL007390


      DOUBLE PRECISION FUNCTION RELLIP (COSP,SINP,K)                        BL007400
C                                                                           BL007410
CCC       RELATIONSHIP FOR DEFINING THE COORDINATE R.                       BL007420
C                                                                           BL007430
      IMPLICIT REAL*8 (A-H,O-Z)                                             BL007440
      COMMON /C01/   A           ,  EPS        ,  JMAX       ,  JMAX1      , BL007450
     1              JMAX2      ,  KMAX       ,  KMAX1,JK     ,   PI        ,BL007460
     2              PSIMAX     ,  S0         ,  S1        ,  UINF       ,   BL007470
     3              WINF       ,  X0         ,  Y0                          BL007480
      COMMON /C02/   ALPHA(100),  DN(50)      ,  DN2(50)    ,  R(100)     , BL007490
     1              RWRTX(100),  S(100)      ,  W(100)    ,  X(100)         BL007500
      COMMON/C023/   A0          ,  B0         ,  C0         ,  P          , BL007510
     1              RB         ,  RC                                        BL007520
      COMMON/C030/   IA, JK2     ,  OTA        ,  PRX        ,  PRXOR      , BL007530
     1              XPOVA                                                   BL007540
      COMMON /EXT/ IBEXT                                                    BL007550
      DIMENSION ZSLO(2)                                                     BL007560
      OTA = COSP*COSP + RB/RC*SINP*SINP                                     BL007570
      IF(IBEXT.EQ.1) GO TO 10                                               BL007580
      RELLIP=(-Y0*COSP+DSQRT((DABS(Y0*COSP))**2.-OTA*(Y0*Y0-(X(K)+X0)     BL007590
     1   *RB*(2.-XPOVA))))/OTA                                              BL007600
      GO TO 20                                                              BL007610
   10 TANP=SINP/COSP                                                        BL007620
      PHI=DATAN(TANP)                                                       BL007630
      XX=X(K)                                                               BL007640
      ZSLO(1)=PHI                                                           BL007650
      CALL SLOALL(XX,ZSLO,1,12,2,0,0,3,0)                                   BL007660
      RELLIP=ZSLO(2)                                                        BL007670
   20 RETURN                                                                BL007680
      END                                                                   BL007690


      SUBROUTINE UXYZ (Y,Z,XPOVA)                                           BL007700
C                                                                           BL007710
```

```
CCC        UXYZ COMPUTES THE VELOCITY COMPONENTS IN THE RECTANGULAR
CCC     COORDINATE DIRECTIONS.
CCC         ** ARGUMENTS **
CCC               Y - Y-COORDINATE.
CCC               Z - Z-COORDINATE.
CCC           XPOVA - XPRIME/A.
C
      IMPLICIT REAL*8 (A-H,O-Z)
      COMMON /C01/   A           , EPS         , JMAX        , JMAX1     ,
     1               JMAX2     , KMAX        , KMAX1,JK    , PI        ,
     2               PSIMAX    , SO          , S1        , UINF       ,
     3               WINF      , XO          , YO
      COMMON /C02/   ALPHA(100),  DN(50)     , DN2(50)     , R(100)    ,
     1               RWRTX(100),  S(100)     , W(100)    , X(100)
      COMMON/C023/   AO          , BO          , CO          , P         ,
     1               RB        , RC
      COMMON/C031/   UXP         , UYP         , UZP
      COMMON/C032/   CU          , CV          , CW
      COMMON /C04/ ALP,COSPH(100),DPHI(100),DPSI,DW(100),PHI(100),PSI,
     1             PSIO,SINPH(100),SINTHB
      COMMON /C050/ K
      COMMON /EXT/ IBEXT
      DIMENSION ZSLO(9)
      IF(IBEXT.EQ.1) GO TO 10
      UAO = 2.*CU/(2.-AO)
      VAO = 2.*CV/(2.-BO)
      WAO = 2.*CW/(2.-CO)
      XPA1 = XPOVA - 1.
      YPORB = (Y + YO) / RB
      ZPORC = Z/RC
      XP2 = XPA1*XPA1
      YP2 = YPORB*YPORB
      ZP2 = ZPORC*ZPORC
      A2P2 = XP2 + YP2 + ZP2
      UXP = (UAO*(A2P2-XP2) - VAO*XPA1*YPORB - WAO*XPA1*ZPORC) / A2P2
      UYP = (VAO*(A2P2-YP2) - UAO*XPA1*YPORB - WAO*YPORB*ZPORC) / A2P2
      UZP = (WAO*(A2P2-ZP2) - UAO*XPA1*ZPORC - VAO*YPORB*ZPORC) / A2P2
      GO TO 20
C
   10 NERR=0
      IF(PSI.LT.0.1001D0 .AND. PSI.GT.0.099D0) NERR=1
      IF(PSI.LT.0.2001D0 .AND. PSI.GT.0.199D0) NERR=1
      IF(PSI.LT.0.3001D0 .AND. PSI.GT.0.299D0) NERR=1
      PI=DARCOS(-1.0D0)
      RAD=PI/180.0D0
      RB2=Y*Y + Z*Z
      YO2=YO*YO
      RBP2=RB2 + 2.0D0*YO*Y + YO2
      RBP=DSQRT(RBP2)
      YP=Y+YO
      YBRP=YP/RBP
```

DATE 04/20/77

```
      PHIP=DARCOS(YBRP)
      PHID=PHIP/RAD
      IF(NERR.EQ.1) WRITE(8,11) XPOVA,PHID,PSI,PHI(K+1)
      ZSLO(1)=PHIP
      CALL SLOALL(XPOVA,ZSLO,1,10,4,0,0,4,NERR)
      UXP=ZSLO(2)*UINF
      UYP=ZSLO(3)*UINF
      UZP=ZSLO(4)*UINF
      IF(NERR.EQ.1) WRITE(8,12) UXP,UYP,UZP
   20 RETURN
   12 FORMAT(10X,3E15.6)
   11 FORMAT(/10X,4E15.6)
      END


      SUBROUTINE GEOM
CC
CC          GEOM SETS UP THE PARAMETERS AND CALLS FOR SOLVING THE VARI
CC     EQUATIONS.  IN DOING THIS A STEP IS TAKEN ALONG THE BODY
CC     (THE PSI-DIRECTION) AND SOLUTIONS FOR ALL PHI AT THIS LOCATIO
CC     ARE FOUND.
C
      IMPLICIT REAL*8 (A-H,O-Z)
       REAL*8 KAPPAS, KAPPAW
      COMMON /C01/ A,EPS,JMAX,JMAX1,JMAX2,KMAX,KMAX1,JK,PI,PSIMAX,SO,S1,
     1UINF,WINF,XO,YO
      COMMON /C02/ ALPHA(100),DN(50),DN2(50),R(100),RWRTX(100),S(100),W(
     1100),X(100)
      COMMON /C03/ F(50,100),FO(50,100),G(50,100),GO(50,100),UE(100),UEO
     1(100),V(50,100),WE(100),WEO(100)
      COMMON /C04/ ALP,COSPH(100),DPHI(100),DPSI,DW(100),PHI(100),PSI,PS
     1IO,SINPH(100),SINTHB
      COMMON /C05/ HPSI(100),HW(100),HWO(100)
      COMMON /C06/ ALBET2,ALBET3,ALP2,BETA1,BETA2,BETA3,BETA4,KAPPAS,KAP
     1PAW
      COMMON /C07/ ALPHO(50),ALPH1(50),ALPH2(50),ALPH3(50),ALPH4(50),ALP
     1H5(50),ALPH6(50)
      COMMON /C010/ LITER,JK1,PI2,SNTHB2
      COMMON /C011/ PIH
      COMMON /C012/ H,HH,WE1
      COMMON /C014/ COSPH2,DW2,PRXS,PRXORS,R2,SINPH2,XPOVAS,X2
      COMMON /C015/ ROLD(100),SOLD(100),THB(100),XOLD(100),XWRTR(100)
      COMMON /C016/ ANGAL,A3,B3,C3,DFDN1,DGDN1,GAMMA(50),TSS
      COMMON /C017/ IPRT,KPRT
      COMMON /C022/ ANGLE(50)
      COMMON /C024/ GPR,PHIPR,RP,UPHP,USP,XP
      COMMON /C026/ XIKA,AL3T
      COMMON /C027/ GOSAV(50),GSAV(50)
```

AD-A051 971   VIRGINIA POLYTECHNIC INST AND STATE UNIV BLACKSBURG --ETC   F/G 20/4
THREE-DIMENSIONAL INCOMPRESSIBLE BOUNDARY LAYERS ON BLUNT BODIE--ETC(U)
MAY 77   D L DWOYER, C H LEWIS, P R GOGINENI

UNCLASSIFIED   VPI/SU-AERO-063-PT-2   NL

2 OF 3
ADA
051971

DATE 04/20/77

```
      COMMON /C030/ IA,JK2,OTA,PRX,PRXOR,XPOVA
      COMMON /C034/ ICON,JTST,NIT
      COMMON /C035/ FTST,FTST1,GTST,GTST1,VTST,VTST1
      COMMON /C036/ ICAP
      COMMON /C037/ FNSAV(50),GNSAV(50)
      COMMON /C041/ CRI,KHALF
      COMMON /C043/ DRDSI,XTST
      ITER=0
      PSIDIF=DPSI-PSIMAX
      W(1)=0.0D0
      DS=0.0D0
      IP=0
      MPRT=0
      NPRT=IPRT-1
      S(1)=-DS
   10 CONTINUE
      NIT=1
      K=0
      ROLD(1)=R(1)
      XOLD(1)=X(1)
      SOLD(1)=S(1)
      PSI=PSI+DPSI
      S(1)=PSI
      IF (PSI.GT.PSIMAX) GO TO 440
      IP=IP+1
      NPRT=NPRT+1
      IF (IP.EQ.1) GO TO 30
      IF (X(1).GE.XTST) GO TO 20
C
C     FIND COORDINATES IN THE PLANE OF SYMMETRY.
C
      CALL COORD (R,X,XWRTR,ROLD,XOLD,2,K,DS)
      GO TO 40
   20 CONTINUE
      CALL COORD (X,R,RWRTX,XOLD,ROLD,1,K,DS)
      GO TO 40
   30 CONTINUE
      R(1)=0.0D0
      X(1)=0.0D0
      HPSI(1)=1.0D0
   40 CONTINUE
      IF (ITER.NE.0) GO TO 70
      CALL CALCV (1,0)
      DO 50 J=1,JMAX
      GAMMA(J)=0.0D0
   50 CONTINUE
      PRINT 450
      DO 60 J=1,JMAX
      PRINT 460, J,F(J,1),G(J,1),V(J,1),ANGLE(J),GAMMA(J)
   60 CONTINUE
   70 CONTINUE
```

```
      IF (IP.NE.1) GO TO 80
      CALL PARVAL (0,IP)
C
C     FIND INITIAL PROFILE SOLUTION FOR F,G, AND V.
C
80    CONTINUE
      DO 90 J=1,JMAX
      FNSAV(J)=F(J,1)
      GNSAV(J)=G(J,1)
90    CONTINUE
      ITER=ITER+1
      IF (IP.NE.1) GO TO 100
      ICON=0
      CALL PSIMOM (IP,0)
      CALL OMEMOM (IP,0)
      ICON=1
      CALL PSIMOM (IP,0)
      CALL OMEMOM (IP,0)
      IF (ITER.LE.LITER) PRINT 470, ITER
      IF (ITER.EQ.LITER) PRINT 480
      CALL PRTPRO (1,0)
100   CONTINUE
      IF (ITER.LT.LITER) GO TO 80
      DS=0.0D0
      DRDSI=(R(1)-ROLD(1))/DPSI
      DO 430 K=1,KMAX
      OMEGA=PHI(K)/PI2
      IF ((IP.NE.1).OR.(K.EQ.1)) GO TO 120
      DO 110 J=1,JMAX
      F(J,K)=F(J,K-1)
      G(J,K)=G(J,K-1)
      V(J,K)=V(J,K-1)
110   CONTINUE
120   CONTINUE
      ICON=0
      NIT=1
      KS=K-1+ICAP
      IF (KS.EQ.0) KS=KS+1
      KM1=K-1
      MPRT=MPRT+1
      IF (K.NE.KMAX) GO TO 230
C
C     FIND F,G, AND V IN THE 180 DEGREE PLANE.
C
130   CONTINUE
      DO 140 J=1,JMAX
      FNSAV(J)=F(J,KMAX)
      GNSAV(J)=G(J,KMAX)
C
C     FIND AVERAGE F AND G VALUES.
C
```

```
140     CONTINUE
        IF (IP.EQ.1) GO TO 160
        IF (NIT.EQ.1) GO TO 180
        DO 150 J=1,JMAX
        F(J,KMAX)=0.5D0*(F(J,KMAX)+FO(J,KS))
        G(J,KMAX)=0.5D0*(G(J,KMAX)+GO(J,KS))
150     CONTINUE
        GO TO 180
160     CONTINUE
        DO 170 J=1,JMAX
        F(J,KMAX)=0.5D0*(F(J,KMAX)+F(J,KMAX1))
        G(J,KMAX)=0.5D0*(G(J,KMAX)+G(J,KMAX1))
170     CONTINUE
180     CONTINUE
        IF (NIT.NE.1) GO TO 190
        CALL PARVAL (KMAX1,IP)
190     CONTINUE
        IF (CRI.EQ.1.) GO TO 200
        IF (IP.NE.1) GO TO 200
        CALL SYMM (KMAX)
        ICON=1
        GO TO 210
200     CONTINUE
        CALL PSIMOM (IP,KMAX1)
        CALL OMEMOM (IP,KMAX1)
        FTST1=F(JTST,KMAX)
        GTST1=G(JTST,KMAX)
        VTST1=V(JTST,KMAX)
        IF (NIT.EQ.1) GO TO 220
        CALL CONV
        ICON=1
        IF (ICON.NE.1) GO TO 220
210     CONTINUE
        CALL PSIMOM (IP,KMAX1)
        CALL OMEMOM (IP,KMAX1)
        ICON=0
        GO TO 410
220     CONTINUE
        NIT=NIT+1
        FTST=FTST1
        GTST=GTST1
        VTST=VTST1
        GO TO 130
C
C       FIND SOLUTION FOR PHI LESS THAN 180 DEGREES.
C
230     CONTINUE
        ROLD(K+1)=R(K+1)
        XOLD(K+1)=X(K+1)
        IF (IP.EQ.1) GO TO 250
C
```

```
C
C      CALCULATE THE COORDINATES AT A STEP AHEAD OF LOCATION WHERE
C      F,G, AND V SOLUTIONS ARE DESIRED.
C
       XPOVAS=XPOVA
       PRXS=PRX
       PRXORS=PRXOR
       IF (X(K+1).GE.XTST) GO TO 240
       CALL COORD (R,X,XWRTR,ROLD,XOLD,2,K,DS)
       GO TO 260
 240   CONTINUE
       CALL COORD (X,R,RWRTX,XOLD,ROLD,1,K,DS)
       GO TO 260
 250   CONTINUE
       R(K+1)=0.0D0
       X(K+1)=0.0D0
       HPSI(K+1)=0.0D0
 260   CONTINUE
       S(K+1)=SOLD(K+1)+DS
       IF ((IP.EQ.1).AND.(K.EQ.1)) GO TO 390
       R2=R(K+1)
       X2=X(K+1)
       DW2=DW(K+1)
       COSPH2=COSPH(K+1)
       SINPH2=SINPH(K+1)
 270   CONTINUE
       DO 280 J=1,JMAX
       FNSAV(J)=F(J,K)
       GNSAV(J)=G(J,K)
       IF ((K.EQ.1).AND.(NIT.EQ.1)) GNSAV(J)=GSAV(J)
 280   CONTINUE
C
C      FIND AVERAGE F AND G VALUES.
C
       IF (IP.EQ.1) GO TO 320
       IF (K.EQ.1) GO TO 290
       IF (NIT.EQ.1) GO TO 340
 290   CONTINUE
       DO 310 J=1,JMAX
       IF ((K.EQ.1).AND.(NIT.EQ.1)) GO TO 300
       F(J,K)=0.0D0
       FO(J,KS)=0.0D0
       G(J,K)=0.0D0
       GO(J,KS)=0.0D0
       F(J,K)=0.5D0*(F(J,K)+FO(J,KS))
       G(J,K)=0.5D0*(G(J,K)+GO(J,KS))
       GO TO 310
 300   CONTINUE
       G(J,K)=GSAV(J)
 310   CONTINUE
       GO TO 340
 320   CONTINUE
```

```
      DO 330 J=1,JMAX
      F(J,K)=0.5D0*(F(J,K)+F(J,K-1))
      G(J,K)=0.5D0*(G(J,K)+G(J,K-1))
330   CONTINUE
340   CONTINUE
      IF (NIT.NE.1) GO TO 350
      CALL PARVAL (KM1,IP)
350   CONTINUE
      IF (CRI.EQ.1.0D0) GO TO 360
      IF ((IP.NE.1).OR.(OMEGA.LE.0.25D0)) GO TO 360
      CALL SYMM (K)
      ICON=1
      GO TO 370
360   CONTINUE
      CALL PSIMOM (IP,KM1)
      CALL OMEMOM (IP,KM1)
      FTST1=F(JTST,K)
      GTST1=G(JTST,K)
      VTST1=V(JTST,K)
      IF (NIT.EQ.1) GO TO 380
      CALL CONV
      ICON=1
      IF (ICON.NE.1) GO TO 380
370   CONTINUE
      CALL PSIMOM (IP,KM1)
      CALL OMEMOM (IP,KM1)
      ICON=0
      GO TO 390
380   CONTINUE
      FTST=FTST1
      GTST=GTST1
      VTST=VTST1
      NIT=NIT+1
      GO TO 270
390   CONTINUE
      IF (KM1.NE.0) GO TO 410
      PSIDIF=PSI+DPSI-PSIMAX
      IF ((NPRT.NE.IPRT).AND.(PSIDIF.LE.0.001D0)) GO TO 400
      MPRT=0
      CALL PRTPRO (2,0)
400   CONTINUE
      GO TO 420
410   CONTINUE
      IF ((MPRT.NE.KPRT).AND.(K.NE.KMAX)) GO TO 420
      MPRT=0
      IF ((NPRT.NE.IPRT).AND.(PSIDIF.LE.0.001D0)) GO TO 420
      CALL PRTPRO (3,KM1)
420   CONTINUE
      IF (K.EQ.KMAX) GO TO 430
      XOLD(K+1)=X(K+1)
      DRDSI=(R(K+1)-ROLD(K+1))/DPSI
```

```
DATE  04/20/77

      ROLD(K+1)=R(K+1)
      SOLD(K+1)=S(K+1)
C
C
430   CONTINUE
      IF (NPRT.EQ.IPRT) NPRT=0
      PSIO=PSI
C
C     GO BACK TO 5 AND TAKE ANOTHER STEP IN THE PSI-DIRECTION.
C
      GO TO 10
440   CONTINUE
      RETURN
C
C
450   FORMAT (1HO,11HITERATION 0//8X,1HJ,5X,1HF,12X,1HG,12X,1HV,10X,5HAL
     1PHA,10X,5HGAMMA)
460   FORMAT (6X,I3,5(E11.4,2X),4(E9.2,2X))
470   FORMAT (1H1,10HITERATION ,I3)
480   FORMAT (30X,15HINITIAL PROFILE)
      END


      SUBROUTINE PRTPRO (LOC,K)                                         BLOO
C                                                                       BLOO
C     CC        SUBROUTINE PRTPRO CONTROLS PRINTING AS THE SOLUTION PROCEBLOO
C     CC        ** ARGUMENTS **                                         BLOO
C     CC             LOC - INDICATES LOCATION ON BODY WHERE PRINTING IS TBLOO
C     CC                   OCCUR.                                       BLOO
C     CC                   = 1, INITIAL PROFILE.                        BLOO
C     CC                   = 3, NOT INITIAL PROFILE.                    BLOO
C     CC               K - SUBSCRIPT IN THE PHI-DIRECTION.              BLOO
C                                                                       BLOO
      IMPLICIT REAL*8 (A-H,O-Z)                                         BLOO
       REAL*8 KAPPAS, KAPPAW                                            BLOO
      COMMON /C01/ A,EPS,JMAX,JMAX1,JMAX2,KMAX,KMAX1,JK,PI,PSIMAX,SO,S1,BLOO
     1UINF,WINF,XO,YO                                                   BLOO
      COMMON /C02/ ALPHA(100),DN(50),DN2(50),R(100),RWRTX(100),S(100),W(BLOO
     1100),X(100)                                                       BLOO
      COMMON /C03/ F(50,100),FO(50,100),G(50,100),GO(50,100),UE(100),UEOBLOO
     1(100),V(50,100),WE(100),WEO(100)                                  BLOO
      COMMON /C04/ ALP,COSPH(100),DPHI(100),DPSI,DW(100),PHI(100),PSI,PSBLOO
     1IO,SINPH(100),SINTHB                                              BLOO
      COMMON /C05/ HPSI(100),HW(100),HWO(100)                           BLOO
      COMMON /C06/ ALBET2,ALBET3,ALP2,BETA1,BETA2,BETA3,BETA4,KAPPAS,KAPBLOO
     1PAW                                                               BLOO
      COMMON /C07/ ALPHO(50),ALPH1(50),ALPH2(50),ALPH3(50),ALPH4(50),ALPBLOO
     1H5(50),ALPH6(50)                                                  BLOO
```

DATE 04/20/77

```
      COMMON /C016/ ANGAL,A3,B3,C3,DFDN1,DGDN1,GAMMA(50),TSS
      COMMON /C022/ ANGLE(50)
      COMMON /C024/ GPR,PHIPR,RP,UPHP,USP,XP
      COMMON /C026/ XIKA,AL3T
      COMMON /C031/ UXP,UYP,UZP
      COMMON /C034/ ICON,JTST,NIT
      COMMON /C044/ TSF
      Y=R(K+1)*COSPH(K+1)
      Z=R(K+1)*SINPH(K+1)
      UINF2=UINF*UINF
      PRCOF=1.0D0-(UE(K+1)**2+WE(K+1)**2)/UINF2
      GO TO (10,30,30), LOC
C
C
C     ****************************INITIAL PROFILE
10    CONTINUE
      WRITE (6,100) PSI,PHI(1),X(1),ALPHA(1),DFDN1,Y,ALBET3,DGDN1,Z,UE(1
     1),WE(1),HPSI(1),R(1),HW(1),S(1),TSS,PRCOF
      DO 20 J=1,JMAX
      WRITE (6,110) J,F(J,1),G(J,1),V(J,1),ANGLE(J),GAMMA(J)
20    CONTINUE
      WRITE (8,90) ALP
      WRITE (8,60)
      WRITE (25) ALP
      GO TO 50
C
C
C     ****************************GENERAL CASE
30    CONTINUE
C     WRITE (6,8000) XP, RP, PHIPR, GPR, USP, UPHP
C     WRITE (6,8001)
C     8001  FORMAT (1H1)
C      FOLLOWING CARDS ARE TO BE USED FOR BLOTTNERS PRESSURE PATCH
C     IF(PSI .GT. 0.491D0 .AND. PSI .LT. 0.505D0) GO TO 31
C     IF(PSI .GT. 0.591D0 .AND. PSI .LT. 0.605D0) GO TO 31
C     IF(PSI .GT. 0.691D0 .AND. PSI. LT. 0.705D0) GO TO 31
C     IF(PSI .GT. 0.791D0 .AND. PSI .LT. 0.805D0) GO TO 31
C     GO TO 50
C     31 IF(PHI(K+1) .LT. 2.2D0) GO TO 50
      KK=K
      IF (KK.EQ.0) KK=1
      WRITE (6,120) PSI,PHI(K+1),X(K+1),ALPHA(K+1),BETA1,DFDN1,Y,ALBET2,
     1BETA2,DGDN1,Z,ALBET3,BETA3,HPSI(K+1),XIKA,BETA4,HW(KK),R(K+1),UE(K
     2+1),KAPPAS,S(K+1),WE(K+1),NIT,KAPPAW,TSS,TSF,ANGAL,PRCOF
C
      UEND=UE(K+1)/UINF
      WEND=WE(K+1)/UINF
      UXPND=UXP/UINF
      UYPND=UYP/UINF
      UZPND=UZP/UINF
      WRITE (25) PSI,HPSI(K+1),HW(KK),R(K+1),S(K+1),UEND,WEND,X(K+1)
      IF (K.EQ.1) WRITE (8,70)                                          BLOO
      WRITE (8,80) PSI,HPSI(K+1),HW(KK),R(K+1),S(K+1),UEND,WEND,UXPND,UYBLOO
     1PND,UZPND                                                         BLOO
C                                                                       BLOO
      IPP=0                                                             BLOO
      IF (IPP.EQ.0) GO TO 50                                            BLOO
      DO 40 J=1,JMAX                                                    BLOO
      WRITE (6,110) J,F(J,K+1),G(J,K+1),V(J,K+1),ANGLE(J),GAMMA(J),ALPH2BLOO
     1(J),ALPH3(J),ALPH4(J),ALPH5(J)                                    BLOO
40    CONTINUE                                                          BLOO
50    CONTINUE                                                          BLOO
      RETURN                                                            BLOO
C     9     8X'J',5X'F',12X'G',12X'V',10X'ALPHA',10X'GAMMA',5X'ALPHA2',  BLOO
C     1     5X'ALPHA3',5X'ALPHA4',5X'ALPHA5')                           BLOO
C                                                                       BLOO
60    FORMAT (//1X,3HPSI,10X,4HHPSI,9X,2HHW,11X,1HR,12X,1HS,12X,9HUPSI/UBLOO
     1INF,4X,7HUW/UINF,6X,7HUX/UINF,6X,7HUY/UINF,6X,7HUZ/UINF/)         BLOO
70    FORMAT (1H1)                                                      BLOO
80    FORMAT (1X,10E13.6)                                               BLOO
90    FORMAT (/10X,11HALPHA(RAD)=,E15.6/)                               BLOO
100   FORMAT (1H0,48X,8HAT XI  , ,E11.4,9H,  PHI = ,E11.4//17X,4HX = ,E11BLOO
     1.4,17X,8HALPHA = ,E11.4,16X,7HDFDN = ,E11.4/17X,4HY = ,E11.4,15X,1BLOO
     20HA-BETA3 = ,E11.4,16X,7HDGDN = ,E11.4/17X,4HZ = ,E11.4,20X,5HUE =BLOO
     3 ,E11.4/52X,5HWE = ,E11.4,16X,7HHPSI = ,E11.4/17X,4HR = ,E11.4,50XBLOO
     4,9HHOMEGA = ,E11.4/17X,4HS = ,E11.4//30X,25HTOTAL  SHEAR  STRESS  BLOO
     5=  ,E11.4,10X,21HPRESSURE COEFFICIENT=,E11.4//8X,1HJ,5X,1HF,12X,1HBLOO
     6G,12X,1HV,10X,5HALPHA,10X,5HGAMMA)                                BLOO
110   FORMAT (6X,I3,5(E11.4,2X),4(E9.2,2X))                             BLOO
120   FORMAT (1H0,48X,8HAT XI = ,E11.4,9H,  PHI = ,E11.4//17X,4HX = ,E11BLOO
     1.4,9X,8HALPHA = ,E11.4,7X,8HBETA1 = ,E11.4,9X,7HDFDN = ,E11.4/17X,BLOO
     25HY =  ,E11.4,7X,10HA-BETA2 = ,E11.4,7X,8HBETA2 = ,E11.4,9X,7HDGDNBLOO
     3 = ,E11.4/17X,4HZ = ,E11.4,7X,10HA-3ETA3 = ,E11.4,7X,8HBETA3 = ,E1BLOO
     41.4,9X,7HHPSI = ,E11.4/42X,7HXIKA = ,E11.4,7X,8HBETA4 = ,E11.4,7X,BLOO
     59HHOMEGA = ,E11.4/17X,4HR = ,E11.4,12X,5HUE = ,E11.4,33X,9HKAPPAS BLOO
     6= ,E11.4/17X,4HS = ,E11.4,12X,5HWE = ,E11.4,9X,6HNIT = ,I2,16X,9HKBLOO
     7APPAW = ,E11.4//30X,25HTOTAL  SHEAR  STRESS  =  ,E11.4,3X,25HTOTALBLOO
     8  SKIN  FRICTION  = ,E11.4/45X,10HANGLE  =  ,E11.4,10X,21HPRESSUREBLOO
     9 COEFFICIENT=,E11.4//)                                            BLOO
      END                                                               BLOO


      SUBROUTINE BODYG                                                  BLOO
C           BODYG   TRANSFORMS BODY AXES TO STAGNATION POINT AXIS       BLOO
C                                                                       BLOO
      IMPLICIT REAL*8(A-H,O-Z)                                          BLOO
      COMMON /CO1/ A,EPS,JMAX,JMAX1,JMAX2,KMAX,KMAX1,JK,PI,PSIMAX,SO,S1,BLOO
     1UINF,WINF,XO,YO                                                   BLOO
      DIMENSION ANYVA(130,5,19), XP(130), P(19), R(19), ADR(130), DRDX(1BLOO
     130)
      CALL ERRSET (207,256,10,0,0,209)
C
      PI=DARCOS(-1.DO)
      DO 10 I=1,150
      READ (10,END=20) XP(I),KPL
      READ (10) ((ANYVA(I,K,J),J=1,KPL),K=1,5)
10    CONTINUE
20    NS=I-1
      DO 60 I=1,NS
      XP(I)=XP(I)-XO
      DO 30 J=1,KPL
      PHP=ANYVA(I,1,J)
      RPR=ANYVA(I,5,J)
      CPHP=DCOS(PHP)
      SPHP=DSIN(PHP)
      Y=RPR*CPHP-YO
      Z=RPR*SPHP
      ZBY=Z/Y
      P(J)=DATAN(ZBY)
      IF (P(J).LT.0.DO) P(J)=P(J)+PI
      R(J)=DSQRT(Y*Y+Z*Z)
30    CONTINUE
      DO 50 J=1,KPL
      PP=ANYVA(I,1,J)
      JJ=0
40    JJ=JJ+1
      IF (P(JJ).LT.PP) GO TO 40
      IF (JJ.LT.3) JJ=3
      IF (JJ.GT.(KPL-1)) JJ=(KPL-1)
      J2=JJ-2
      J1=JJ-1
      JP=JJ+1
      CALL INTER4 (PP,P(J2),P(J1),P(JJ),P(JP),R(J2),R(J1),R(JJ),R(JP),RR
     1)
      WRITE (8,120) P(J2),P(J1),P(JJ),P(JP),R(J2),R(J1),R(JJ),R(JP)
      WRITE (8,130) I,J,JJ,PP,RR
      ANYVA(I,2,J)=RR
50    CONTINUE
      WRITE (12) XP(I),KPL
      WRITE (12) ((ANYVA(I,K,J),J=1,KPL),K=1,2)
      WRITE (6,90) I,XP(I),KPL
      WRITE (6,100) (J,ANYVA(I,1,J),ANYVA(I,2,J),J=1,KPL)
60    CONTINUE
      WRITE (6,110)
      DO 80 J=1,KPL
      DO 70 I=1,NS
70    ADR(I)=ANYVA(I,2,J)
      CALL DERIV5 (XP,ADR,NS,0,0,DRDX)
      WRITE (13) ANYVA(1,1,J),NS
      WRITE (13) (ANYVA(I,2,J),I=1,NS),(XP(I),I=1,NS),(DRDX(I),I=1,NS)
      WRITE (6,90) J,ANYVA(1,1,J),NS                                  BLOO  59
      WRITE (6,100) (I,ANYVA(I,2,J),XP(I),DRDX(I),I=1,NS)             BLOO  60
80    CONTINUE                                                        BLOO  61
      REWIND 12                                                       BLOO  62
      REWIND 13                                                       BLOO  63
      RETURN                                                          BLOO  64
C                                                                     BLOO  65
90    FORMAT (//1X,I3,6X,E15.6,5X,I3)                                 BLOO  66
100   FORMAT (/1X,4(I3,2X,2E13.6))                                    BLOO  67
110   FORMAT (//20X,8HUNIT  13//)                                     BLOO  68
120   FORMAT (5E13.6,2X,5E13.6)                                       BLOO  69
130   FORMAT (3I6,5X,5E15.6)                                          BLOO  70
      END                                                             BLOO  71


      SUBROUTINE DERIV5 (X,Y,NP,JP,IP,DYDX)                           BLOO   1
      IMPLICIT REAL *8(A-H,O-Z)                                       BLOO   2
C                                                                     BLOO   3
      DIMENSION X(130), Y(130), DYDX(130)                             BLOO   4
C                                                                     BLOO   5
      DO 10 J=1,NP                                                    BLOO   6
      K=J                                                             BLOO   7
      IF (IP.EQ.1) K=JP                                               BLOO   8
      IF (K.LT.3) K=3                                                 BLOO   9
      IF (K.GT.(NP-2)) K=NP-2                                         BLOO  10
      XX=X(J)                                                         BLOO  11
      XX1=X(K-2)                                                      BLOO  12
      XX2=X(K-1)                                                      BLOO  13
      XX3=X(K)                                                        BLOO  14
      XX4=X(K+1)                                                      BLOO  15
      XX5=X(K+2)                                                      BLOO  16
      YY1=Y(K-2)                                                      BLOO  17
      YY2=Y(K-1)                                                      BLOO  18
      YY3=Y(K)                                                        BLOO  19
      YY4=Y(K+1)                                                      BLOO  20
      YY5=Y(K+2)                                                      BLOO  21
      CALL FD5 (XX,XX1,XX2,XX3,XX4,XX5,YY1,YY2,YY3,YY4,YY5,DER)       BLOO  22
      DYDX(J)=DER                                                     BLOO  23
      IF (IP.EQ.1) GO TO 20                                           BLOO  24
10    CONTINUE                                                        BLOO  25
20    RETURN                                                          BLOO  26
      END                                                             BLOO  27


      SUBROUTINE FD5 (X,X1,X2,X3,X4,X5,F1,F2,F3,F4,F5,FX)             BLOO   1
```

DATE 04/20/77

```
      IMPLICIT REAL*8(A-H,O-Z)                                          BLOO   2
      A1=(X-X4)*(X-X5)*(2.D0*X-X2-X3)+(X-X2)*(X-X3)*(2.D0*X-X4-X5)      BLOO   3
      A2=(X-X4)*(X-X5)*(2.D0*X-X1-X3)+(X-X1)*(X-X3)*(2.D0*X-X4-X5)      BLOO   4
      A3=(X-X4)*(X-X5)*(2.D0*X-X1-X2)+(X-X1)*(X-X2)*(2.D0*X-X4-X5)      BLOO   5
      A4=(X-X3)*(X-X5)*(2.D0*X-X1-X2)+(X-X1)*(X-X2)*(2.D0*X-X3-X5)      BLOO   6
      A5=(X-X3)*(X-X4)*(2.D0*X-X1-X2)+(X-X1)*(X-X2)*(2.D0*X-X3-X4)      BLOO   7
      D1=(X1-X2)*(X1-X3)*(X1-X4)*(X1-X5)                                BLOO   8
      D2=(X2-X1)*(X2-X3)*(X2-X4)*(X2-X5)                                BLOO   9
      D3=(X3-X1)*(X3-X2)*(X3-X4)*(X3-X5)                                BLOO  10
      D4=(X4-X1)*(X4-X2)*(X4-X3)*(X4-X5)                                BLOO  11
      D5=(X5-X1)*(X5-X2)*(X5-X3)*(X5-X4)                                BLOO  12
      C1=A1/D1                                                          BLOO  13
      C2=A2/D2                                                          BLOO  14
      C3=A3/D3                                                          BLOO  15
      C4=A4/D4                                                          BLOO  16
      C5=A5/D5                                                          BLOO  17
      FX=C1*F1+C2*F2+C3*F3+C4*F4+C5*F5                                  BLOO  18
      RETURN                                                            BLOO  19
      END                                                               BLOO  20


      SUBROUTINE INTER4 (X,X1,X2,X3,X4,F1,F2,F3,F4,F)                   BLOO   1
      IMPLICIT REAL*8(A-H,O-Z)                                          BLOO   2
      A1=(X-X2)*(X-X3)*(X-X4)                                           BLOO   3
      A2=(X-X1)*(X-X3)*(X-X4)                                           BLOO   4
      A3=(X-X1)*(X-X2)*(X-X4)                                           BLOO   5
      A4=(X-X1)*(X-X2)*(X-X3)                                           BLOO   6
      D1=(X1-X2)*(X1-X3)*(X1-X4)                                        BLOO   7
      D2=(X2-X1)*(X2-X3)*(X2-X4)                                        BLOO   8
      D3=(X3-X1)*(X3-X2)*(X3-X4)                                        BLOO   9
      D4=(X4-X1)*(X4-X2)*(X4-X3)                                        BLOO  10
      C1=A1/D1                                                          BLOO  11
      C2=A2/D2                                                          BLOO  12
      C3=A3/D3                                                          BLOO  13
      C4=A4/D4                                                          BLOO  14
      F=C1*F1+C2*F2+C3*F3+C4*F4                                         BLOO  15
      RETURN                                                            BLOO  16
      END                                                               BLOO  17


      SUBROUTINE SLOALL (XX,Z,I1,IT,NV,MX,MZ,NOR,NPRT)                  BLOO   1
      IMPLICIT REAL*8(A-H,O-Z)                                          BLOO   2
C                                                                       BLOO   3
C      LATEST REVISION DONE ON 11/23/1976   BY   P.R. GOGINENI          BLOO   4
C                                                                       BLOO   5
```

```
C
C     EACH CALL TO THE SUBROUTINE SLOW RETURNS A VALUE OF THE ARRAYS
C     Z(JI) CORRESPONDING TO XX,Z(II).  THE VALUES ARE OBTAINED
C     BY DOUBLE INTERPOLATION FROM THE DATA READ FROM A TAPE
C
C      DESCRIPTION OF ARGUMENTS
C
C     XX,Z(II)  ARE THE INPUT VALUES ; II   IDENTIFIES THE ARRAY FROM TH
C           Z ARRAYS (NV) WRITTEN ON THE DATA TAPE
C
C     IT        UNIT NO.(DATA TAPE)  FROM WHICH DATA HAS TO BE READ
C
C     NV        NO. OF ARRAYS WRITTEN ON DATA TAPE AS SECOND RECORD
C     MX=0      XX- VALUES MONOTONICALLY INCREASING
C     MX=1      XX-    :        :       DECREASING
C     MZ        SAME AS ABOVE  BUT FOR Z-VALUES
C     NOR       ORDER OF THE LEGENDRE POLYNOMIAL TO BE USED
C            FOR INTERPOLATION; EX. NOR=2 (2ND ORDER) USES 3 POINTS
C
C     NPRT=1    DIAGNOSTIC PRINTOUT
C      =0    NO     :        :
C
      NDUMMY=0
C
C     FOR EACH  XX ,  DATA ON UNIT  IT,  IS AS SHOWN BELOW
C
C     FIRST RECORD:    XX,JJ
C     SECOND  :       1  Z(1), Z(2), . . ., Z(NV)
C                   2  Z(1), Z(2), . . ., Z(NV)
C                   .  .     .     . . .   .
C                   .  .     .     . . .   .
C                  JJ  Z(1), Z(2), . . ., Z(NV)
C

      COMMON /COMIJK/ IJKEDG,IJK
      COMMON /COMSLO/ X(5),Y(5,5,120),Y2(5,120),XX1,JJ
      DIMENSION Z(5)
C
C
      IF (NOR.LE.3) GO TO 10
      MMI=NOR-1
      MMI1=2
      NCI=2
      NCI1=4
      GO TO 20
10    MMI=NOR
      MMI1=1
      NCI=1
      NCI1=3
      IF (NOR.EQ.3) NCI=2
20    NOR1=NOR+1
```

```
      NOR2=NOR+2
      IF (NOR.EQ.3) NOR2=2*NOR
      IF (IJK.NE.301) XX1=10.0**20
      IF (IJK.NE.301) GO TO 50
      IF (MX.EQ.0) GO TO 30
      IF (XX.GT.X(NCI)) GO TO 50
      IF (XX.GE.X(NCI1)) GO TO 110
      GO TO 40
30    IF (XX.LT.X(NCI)) GO TO 50
      IF (XX.LE.X(NCI1)) GO TO 110
40    BACKSPACE IT
      BACKSPACE IT
      GO TO 60
50    REWIND IT
      IJK=301
60    READ (IT) X1,JJ
      READ (IT)
      IF (MX.EQ.0) GO TO 70
      IF (X1.GT.XX) GO TO 60
      GO TO 80
70    IF (X1.LT.XX) GO TO 60
80    DO 90 K=1,NOR2
      BACKSPACE IT
90    CONTINUE
      DO 100 K=1,NOR1
      READ (IT) X(K),JJ
      READ (IT) ((Y(K,I,L),L=1,JJ),I=1,NV)
100   CONTINUE
110   CONTINUE
      IF (XX.EQ.XX1) GO TO 140
      DO 130 I=1,NV
      DO 120 L=1,JJ
      IF (NPRT.EQ.1) WRITE (11,240) (X(K),Y(K,I,L),K=1,NOR1)
      IF (NOR.EQ.2) CALL INTER3 (XX,X(1),X(2),X(3),Y(1,I,L),Y(2,I,L),Y(3
     1,I,L),Y2(I,L))
      IF (NOR.EQ.3) CALL INTER4 (XX,X(1),X(2),X(3),X(4),Y(1,I,L),Y(2,I,L
     1),Y(3,I,L),Y(4,I,L),Y2(I,L))
      IF (NOR.EQ.4) CALL INTER5 (XX,X(1),X(2),X(3),X(4),X(5),Y(1,I,L),Y(
     12,I,L),Y(3,I,L),Y(4,I,L),Y(5,I,L),Y2(I,L))
      IF (NPRT.EQ.1) WRITE (11,250) XX,Y2(I,L)
120   CONTINUE
130   CONTINUE
140   CONTINUE
      MM=0
150   MM=MM+1
      IF (MZ.EQ.0) GO TO 160
      IF (Z(I1).LT.Y2(I1,MM)) GO TO 150
      GO TO 170
160   IF (Z(I1).GT.Y2(I1,MM)) GO TO 150
170   IF (Y2(1,MM).LT.0.0D0) GO TO 220
      IF (MM.LT.MMI) MM=MMI
```

```
        IF (MM.GT.(JJ-MMI1)) MM=JJ-MMI1                                   BLOO 108
        MM1=MM-1                                                          BLOO 109
        MM2=MM-2                                                          BLOO 110
        MP1=MM+1                                                          BLOO 111
        MP2=MM+2                                                          BLOO 112
        IF (NPRT.EQ.0) GO TO 180                                          BLOO 113
        IF (NOR.EQ.2) WRITE (11,260) Z(I1),(Y2(I1,MR),MR=MM1,MP1)         BLOO 114
        IF (NOR.EQ.3) WRITE (11,260) Z(I1),(Y2(I1,MR),MR=MM2,MP1)         BLOO 115
        IF (NOR.EQ.4) WRITE (11,260) Z(I1),(Y2(I1,MR),MR=MM2,MP2)         BLOO 116
180     DO 230 J1=1,NV                                                    BLOO 117
        IF (J1.EQ.I1) GO TO 230                                           BLOO 118
        IF (NOR.EQ.3) GO TO 190                                           BLOO 119
        IF (NOR.EQ.4) GO TO 200                                           BLOO 120
        IF (NPRT.EQ.1) WRITE (11,270) J1,(Y2(J1,MR),MR=MM1,MP1)           BLOO 121
        CALL INTER3 (Z(I1),Y2(I1,MM1),Y2(I1,MM),Y2(I1,MP1),Y2(J1,MM1),Y2(JBLOO 122
       11,MM),Y2(J1,MP1),Z(J1))                                           BLOO 123
        GO TO 210                                                         BLOO 124
190     IF (NPRT.EQ.1) WRITE (11,270) J1,(Y2(J1,MR),MR=MM2,MP1)           BLOO 125
        CALL INTER4 (Z(I1),Y2(I1,MM2),Y2(I1,MM1),Y2(I1,MM),Y2(I1,MP1),Y2(JBLOO 126
       11,MM2),Y2(J1,MM1),Y2(J1,MM),Y2(J1,MP1),Z(J1))                     BLOO 127
        GO TO 210                                                         BLOO 128
200     IF (NPRT.EQ.1) WRITE (11,270) J1,(Y2(J1,MR),MR=MM2,MP2)           BLOO 129
        CALL INTER5 (Z(I1),Y2(I1,MM2),Y2(I1,MM1),Y2(I1,MM),Y2(I1,MP1),Y2(IBLOO 130
       11,MP2),Y2(J1,MM2),Y2(J1,MM1),Y2(J1,MM),Y2(J1,MP1),Y2(J1,MP2),Z(J1BLOO 131
       2)                                                                 BLOO 132
210     IF (NPRT.EQ.1) WRITE (11,280) Z(I1),Z(J1),J1                      BLOO 133
        XX1=XX                                                            BLOO 134
        GO TO 230                                                         BLOO 135
220     CONTINUE                                                          BLOO 136
        Z(J1)=Y2(J1,MM)-(Y2(J1,MM)-Y2(J1,MM1))/(Y2(I1,MM)-Y2(I1,MM1))*(Y2(BLOO 137
       1I1,MM)-Z(I1))                                                     BLOO 138
        XX1=XX                                                            BLOO 139
230     CONTINUE                                                          BLOO 140
        RETURN                                                            BLOO 141
C                                                                         BLOO 142
240     FORMAT (1X,5(2E13.6))                                             BLOO 143
250     FORMAT (10X,E15.6,15X,E15.6)                                      BLOO 144
260     FORMAT (10X,E15.6,5X,5E15.6)                                      BLOO 145
270     FORMAT (10X,I3,5X,5E15.6)                                         BLOO 146
280     FORMAT (10X,E15.6,5X,E15.6,2X,I3)                                 BLOO 147
        END                                                               BLOO 148


        SUBROUTINE INTER5 (X,X1,X2,X3,X4,X5,F1,F2,F3,F4,F5,F)             BLOO   1
        IMPLICIT REAL*8(A-H,O-Z)                                          BLOO   2
C                                                                         BLOO   3
C       SUBROUTINE INTER5 IS CALLED BY SUBROUTINES BLUNT1, BLUNT2, EDGE,  BLOO   4
C       INIT, MACH, SLOW, WALL, ZRO, AND INTRP5.                          BLOO   5
```

```
DATE  04/20/77

C
C     SUBROUTINE INTER5 INTERPOLATES FOR THE VALUE F CORRESPONDING TO
C     POINT X USING 5 POINT LAGRANGIAN INTERPOLATION FORMULA.
C
C     ASSUMES X1 .LE. X .LE. X5.
C
      A1=(X-X2)*(X-X3)*(X-X4)*(X-X5)
      A2=(X-X1)*(X-X3)*(X-X4)*(X-X5)
      A3=(X-X1)*(X-X2)*(X-X4)*(X-X5)
      A4=(X-X1)*(X-X2)*(X-X3)*(X-X5)
      A5=(X-X1)*(X-X2)*(X-X3)*(X-X4)
      D1=(X1-X2)*(X1-X3)*(X1-X4)*(X1-X5)
      D2=(X2-X1)*(X2-X3)*(X2-X4)*(X2-X5)
      D3=(X3-X1)*(X3-X2)*(X3-X4)*(X3-X5)
      D4=(X4-X1)*(X4-X2)*(X4-X3)*(X4-X5)
      D5=(X5-X1)*(X5-X2)*(X5-X3)*(X5-X4)
      C1=A1/D1
      C2=A2/D2
      C3=A3/D3
      C4=A4/D4
      C5=A5/D5
      F=C1*F1+C2*F2+C3*F3+C4*F4+C5*F5
      RETURN
      END


      SUBROUTINE INTER3 (X,X1,X2,X3,F1,F2,F3,F)
      IMPLICIT REAL*8(A-H,O-Z)
C
C     SUBROUTINE INTER3 IS CALLED BY SUBROUTINE ADDETA.
C
C     SUBROUTINE INTER3 INTERPOLATES FOR THE VALUE F CORRESPONDING TO
C     POINT X USING 3 POINT LAGRANGIAN INTERPOLATION.
C
C     ASSUMES X1 .LE. X .LE. X3.
C
      A1=(X-X2)*(X-X3)
      A2=(X-X1)*(X-X3)
      A3=(X-X1)*(X-X2)
      D1=(X1-X2)*(X1-X3)
      D2=(X2-X1)*(X2-X3)
      D3=(X3-X1)*(X3-X2)
      C1=A1/D1
      C2=A2/D2
      C3=A3/D3
      F=C1*F1+C2*F2+C3*F3
      RETURN
      END
      SUBROUTINE INTER4 (X,X1,X2,X3,X4,F1,F2,F3,F4,F)                    BLOO   1
      IMPLICIT REAL*8(A-H,O-Z)                                           BLOO   2
      A1=(X-X2)*(X-X3)*(X-X4)                                            BLOO   3
      A2=(X-X1)*(X-X3)*(X-X4)                                            BLOO   4
      A3=(X-X1)*(X-X2)*(X-X4)                                            BLOO   5
      A4=(X-X1)*(X-X2)*(X-X3)                                            BLOO   6
      D1=(X1-X2)*(X1-X3)*(X1-X4)                                         BLOO   7
      D2=(X2-X1)*(X2-X3)*(X2-X4)                                         BLOO   8
      D3=(X3-X1)*(X3-X2)*(X3-X4)                                         BLOO   9
      D4=(X4-X1)*(X4-X2)*(X4-X3)                                         BLOO  10
      C1=A1/D1                                                           BLOO  11
      C2=A2/D2                                                           BLOO  12
      C3=A3/D3                                                           BLOO  13
      C4=A4/D4                                                           BLOO  14
      F=C1*F1+C2*F2+C3*F3+C4*F4                                          BLOO  15
      RETURN                                                             BLOO  16
      END                                                                BLOO  17


C       THESE SUBROUTINES ARE DUMMIED TO GET THE INVISCID PRESSURE       BLOO   1
C       DISTRIBUTION FROM  BLOTTNERS PROGRAM                             BLOO   2
C                                                                        BLOO   3
      SUBROUTINE AVERG (K,IP)                                            BLOO   4
      RETURN                                                             BLOO   5
      END                                                                BLOO   6
      SUBROUTINE CALCV (IP,K)                                            BLOO   1
      RETURN                                                             BLOO   2
      END                                                                BLOO   3
      SUBROUTINE CONV                                                    BLOO   1
      RETURN                                                             BLOO   2
      END                                                                BLOO   3
      SUBROUTINE OMEMOM (IP,K)                                           BLOO   1
      RETURN                                                             BLOO   2
      END                                                                BLOO   3
      SUBROUTINE PSIMOM (IP,K)                                           BLOO   1
      RETURN                                                             BLOO   2
      END                                                                BLOO   3
      SUBROUTINE SYMM (K)                                                BLOO   1
      RETURN                                                             BLOO   2
      END                                                                BLOO   3
      SUBROUTINE TRIDAG (W,WO,K,IP,WN)                                   BLOO   1
      RETURN                                                             BLOO   2
      END                                                                BLOO   3
```

DATE 04/20/77

```
      SUBROUTINE DPRX (COSP,RR,PRX)                                     BLOO   1
      IMPLICIT REAL*8(A-H,O-Z)                                          BLOO   2
      DIMENSION ZSLO(3)                                                 BLOO   3
      PHI=DARCOS(COSP)                                                  BLOO   4
      XX=PHI                                                            BLOO   5
      ZSLO(1)=RR                                                        BLOO   6
      CALL SLOALL (XX,ZSLO,1,13,3,0,0,3,0)                              BLOO   7
      PRX=ZSLO(3)                                                       BLOO   8
      RETURN                                                            BLOO   9
      END                                                               BLOO  10
```

2. Program DERVAT Listing

```
DATE  05/13/77

C       PROGRAM MAIN
        IMPLICIT REAL*8(A-H,O-Z)
        DIMENSION ANYVA(100,14,40), XS(100), ADUM(100), DADUM(100)
        CALL ERRSET (207,256,10,0,0,209)
C
C       THIS PROGRAM COMPUTES DERIVATIVES OF THE VELOCITY COMPONENTS, BODY
C       RADIUS AND SCALE FACTORS.  THE RAW DATA IS READ FROM UNIT 25. THE
C       X DERIVATIVES ARE COMPUTED USING A 5 POINT LAGRANGIAN POLYNOMIAL
C       IN SUBROUTINES DERIV5 AND FD5. THE ORIGINAL DATA ALONG WITH THE
C       REQUIRED DERIVATIVES ARE WRITTEN ON UNIT 26
C
        I=1
        J=0
        READ (25) ALPHA
 10     J=J+1
        READ (25) XS(J),(ANYVA(I,K,J),K=1,7)
        IF (J.EQ.1) GO TO 10
        ANYVA(I,1,J)=ANYVA(I,1,1)
        XSD=XS(J)-XS(J-1)
        IF (XSD.GT.0.1D-02) GO TO 20
        GO TO 10
 20     BACKSPACE 25
        J=J-1
        KPL=J
        XS(I)=XS(J)
        I=I+1
        WRITE (6,200) ALPHA,KPL
 30     CONTINUE
        DO 40 J=1,KPL
 40     READ (25,END=50) XS(I),(ANYVA(I,K,J),K=1,7)
        I=I+1
        GO TO 30
 50     NS=I-1
        WRITE (6,170)
        DO 60 I=1,NS
        WRITE (6,180) I,XS(I),(ANYVA(I,K,1),K=1,7)
 60     WRITE (6,190) ((ANYVA(I,K,J),K=1,7),J=2,KPL)
        DO 100 J=1,KPL
        DO 90 K=1,7
        IF (K.EQ.1) K1=12
        IF (K.EQ.2) K1=13
        IF (K.EQ.3) K1=10
        IF (K.EQ.4) GO TO 90
        IF (K.EQ.5) K1=8
        IF (K.EQ.6) K1=9
        IF (K.EQ.7) K1=14
        DO 70 I=1,NS
 70     ADUM(I)=ANYVA(I,K,J)
        CALL DERIV5 (XS,ADUM,NS,0,0,DADUM)
        DO 80 I=1,NS
 80     ANYVA(I,K1,J)=DADUM(I)
```

```
90     CONTINUE
100    CONTINUE
       DO 130 J=1,KPL
       DO 110 I=1,NS
110    ADUM(I)=ANYVA(I,10,J)
       CALL DERIV5 (XS,ADUM,NS,0,0,DADUM)
       DO 120 I=1,NS
120    ANYVA(I,11,J)=DADUM(I)
130    CONTINUE
       WRITE (26) ALPHA,KPL
       WRITE (30,200) ALPHA,KPL
       DO 140 I=1,NS
       WRITE (26) I,XS(I),((ANYVA(I,K,J),K=1,14),J=1,KPL)
140    CONTINUE
       WRITE (30,210)
       WRITE (29,220)
       DO 160 I=1,NS
       WRITE (30,230) I,XS(I),ANYVA(I,1,1),ANYVA(I,12,1),ANYVA(I,2,1),ANY
      1VA(I,13,1),ANYVA(I,3,1),ANYVA(I,10,1),ANYVA(I,11,1)
       WRITE (29,230) I,XS(I),ANYVA(I,4,1),ANYVA(I,7,1),ANYVA(I,14,1),ANY
      1VA(I,5,1),ANYVA(I,8,1),ANYVA(I,6,1),ANYVA(I,9,1)
       DO 150 J=2,KPL
       WRITE (30,240) ANYVA(I,1,J),ANYVA(I,12,J),ANYVA(I,2,J),ANYVA(I,13,
      1J),ANYVA(I,3,J),ANYVA(I,10,J),ANYVA(I,11,J)
       WRITE (29,240) ANYVA(I,4,J),ANYVA(I,7,J),ANYVA(I,14,J),ANYVA(I,5,J
      1),ANYVA(I,8,J),ANYVA(I,6,J),ANYVA(I,9,J)
150    CONTINUE
160    CONTINUE
       STOP
C
170    FORMAT (/5X,4HISTA,4X,2HXS,13X,4HHPSI,11X,2HHW,13X,1HR,14X,1HS,14X
      1,4HUPSI,11X,2HUW,13X,4HXBLT/)
180    FORMAT (5X,I3,2X,8E15.6)
190    FORMAT (25X,7E15.6)
200    FORMAT (/10X,11HALPHA(RAD)=,E15.6,5X,14HNO. OF PLANES=,I3/)
210    FORMAT (/5X,4HISTA,4X,2HXS,13X,4HHPSI,11X,7HD(HPSI),8X,2HHW,13X,5H
      1D(HW),10X,1HR,14X,4HD(R),11X,5HD2(R)/)
220    FORMAT (/5X,4HISTA,4X,2HXS,13X,1HS,14X,1HX,14X,4HD(X),11X,4HUPSI,1
      11X,7HD(UPSI),8X,2HUW,13X,5HD(UW)/)
230    FORMAT (5X,I3,2X,8E15.6)
240    FORMAT (25X,7E15.6)
       END


       SUBROUTINE DERIV5 (X,Y,NP,JP,IP,DYDX)
       IMPLICIT REAL *8(A-H,O-Z)
C
       DIMENSION X(50), Y(50), DYDX(50)
```

```
C
C     THIS SUBROUTINE SETS UP THE FITTING OF FIFTH ORDER LAGRANGIAN
C     POLYNOMIALS TO DATA IN ARRAYS X AND Y.  THE ACTUAL CALCUTATIONS
C     ARE CARRIED OUT IN SUBROUTINE FD5.
C
      DO 10 J=1,NP
      K=J
      IF (IP.EQ.1) K=JP
      IF (K.LT.3) K=3
      IF (K.GT.(NP-2)) K=NP-2
      XX=X(J)
      XX1=X(K-2)
      XX2=X(K-1)
      XX3=X(K)
      XX4=X(K+1)
      XX5=X(K+2)
      YY1=Y(K-2)
      YY2=Y(K-1)
      YY3=Y(K)
      YY4=Y(K+1)
      YY5=Y(K+2)
      CALL FD5 (XX,XX1,XX2,XX3,XX4,XX5,YY1,YY2,YY3,YY4,YY5,DER)
      DYDX(J)=DER
      IF (IP.EQ.1) GO TO 20
   10 CONTINUE
   20 RETURN
      END


      SUBROUTINE FD5 (X,X1,X2,X3,X4,X5,F1,F2,F3,F4,F5,FX)
      IMPLICIT REAL*8(A-H,O-Z)
C
C     THIS SUBROUTINE FITS DATA AND CALCUTATES ITS FIRST DERIVATIVE
C     USING A FIFTH ORDER POLYNOMIAL
C
      A1=(X-X4)*(X-X5)*(2.D0*X-X2-X3)+(X-X2)*(X-X3)*(2.D0*X-X4-X5)
      A2=(X-X4)*(X-X5)*(2.D0*X-X1-X3)+(X-X1)*(X-X3)*(2.D0*X-X4-X5)
      A3=(X-X4)*(X-X5)*(2.D0*X-X1-X2)+(X-X1)*(X-X2)*(2.D0*X-X4-X5)
      A4=(X-X3)*(X-X5)*(2.D0*X-X1-X2)+(X-X1)*(X-X2)*(2.D0*X-X3-X5)
      A5=(X-X3)*(X-X4)*(2.D0*X-X1-X2)+(X-X1)*(X-X2)*(2.D0*X-X3-X4)
      D1=(X1-X2)*(X1-X3)*(X1-X4)*(X1-X5)
      D2=(X2-X1)*(X2-X3)*(X2-X4)*(X2-X5)
      D3=(X3-X1)*(X3-X2)*(X3-X4)*(X3-X5)
      D4=(X4-X1)*(X4-X2)*(X4-X3)*(X4-X5)
      D5=(X5-X1)*(X5-X2)*(X5-X3)*(X5-X4)
      C1=A1/D1
      C2=A2/D2
      C3=A3/D3
      C4=A4/D4
      C5=A5/D5
      FX=C1*F1+C2*F2+C3*F3+C4*F4+C5*F5
      RETURN
      END
```

3. Program INVTAP Listing

```
DATE  05/13/77

C        PROGRAM INVTAP CREATES THE INVISCID DATA TAPE AS REQUIRED
C        BY THE BOUNDARY LAYER PROGRAMME; CALLING  DISKIN
      IMPLICIT REAL*8(A-H,O-Z)
C
      WRITE (6,10)
      CALL ERRSET (207,256,10,0,0,209)
C
      CALL DISKIN
      STOP
C
10    FORMAT (/10X,60HGENERATING INVISCID DATA TAPE FOR BOUNDARY LAYER
     1   PROGRAM//)
      END
      SUBROUTINE DISKIN
      IMPLICIT REAL*8 (A-H,O-Z)
      REAL*8 NOSE
      DIMENSION RADS(40), US(40), VS(40), XSURFS(40), HXS(40), HWS(40),
     1USX(40), VSX(40), RADSX(40), RADSXX(40), HXSX(40), HWSX(40), X(40)
     2, XX(40), ARADS(40), AUS(40), AVS(40), AHXS(40), AHWS(40), AXSRFS(
     340), AUSX(40), AVSX(40), ARADSX(40), ARDSXX(40), AHXSX(40), AHWSX(
     440), AX(40), AXX(40), FLOFLD(14,40)
      DATA BLUNT,SHARP/5HBLUNT,5HSHARP/
      ICOUNT=0
      WRITE (30,250)
      WRITE (30,140)
      WRITE (30,140)
      READ (25) ALPHA,KL
C
C     READ THE FIRST SET OF INVISCID AND BODY DATA
C
      READ (25) ISTA,XS,((FLOFLD(J,K),J=1,14),K=1,KL)
C
C     XS = LONGITUDINAL SURFACE COORDINATE
C     FLOFLD(1,K) = STREAMWISE SCALE FACTOR
C     FLOFLD(2,K) = CROSS SCALE FACTOR
C     FLOFLD(3,K) = RADIUS
C     FLOFLD(4,K) = SURFACE DISTANCE FROM NOSE
C     FLOFLD(5,K) = STREAMWISE VELOCITY COMPONENT
C     FLOFLD(6,K) = CROSS FLOW VELOCITY COMPONENT
C     FLOFLD(7,K) = AXIAL DISTANCE
C     FLOFLD(8,K) = XS DERIVATIVE OF STREAMWISE VELOCITY COMPONENT
C     FLOFLD(9,K) = XS DERIVATIVE OF CROSS FLOW VELOITY COMPONENT
C     FLOFLD(10,K) = XS DERIVATIVE OF BODY RADIUS
C     FLOFLD(11,K) = 2ND X DERIVATIVE OF BODY RADIUS
C     FLOFLD(12,K) = XS DERIVATIVE OF STREAMWISE SCALE FACTOR
C     FLOFLD(13,K) = XS DERIVATIVE OF CROSS SCALE FACTOR
C     FLOFLD(14,K) = XS DERIVATIVE OF AXIAL DISTANCE
C     KL IS THE NUMBER OF PLANES
C     K=1 IS THE WINDWARD PLANE
C     K=KL IS THE LEEWARD PLANE
C
```

DATE 05/13/77

```
      PI=DARCOS(-1.0D0)
      WRITE (30,140)
      WRITE (30,150) KL
      WRITE (30,140)
      ALPH=ALPHA*(180.0D0/PI)
      WRITE (30,160) ALPH
      WRITE (30,140)
C
C     UNIT 10 IS THE EDGE PROPERTY DATA SET
C
      WRITE (10) ALPH,KL
      WRITE (30,140)
      WRITE (30,240)
      WRITE (30,140)
C
C
      WRITE (30,170) ISTA,XS
      WRITE (30,140)
      WRITE (30,180)
      DO 10 K=1,KL
      WRITE (30,200) K,(FLOFLD(I,K),I=1,7)
10    CONTINUE
      WRITE (30,190)
      DO 20 K=1,KL
20    WRITE (30,200) K,(FLOFLD(I,K),I=8,14)
      WRITE (30,140)
      GO TO 60
30    CONTINUE
C
C     FLOWFIELD DATA FROM THE INVISCID SOLUTION IS READ FROM UNIT 25
C
      READ (25,END=130) ISTA,XS,((FLOFLD(J,K),J=1,14),K=1,KL)
C
      FLOFLD(6,1)=0.0D0
      WRITE (30,170) ISTA,XS
      WRITE (30,140)
      WRITE (30,180)
      DO 40 K=1,KL
      WRITE (30,200) K,(FLOFLD(I,K),I=1,7)
40    CONTINUE
      WRITE (30,190)
      DO 50 K=1,KL
50    WRITE (30,200) K,(FLOFLD(I,K),I=8,14)
      WRITE (30,140)
60    CONTINUE
      DO 70 K=1,KL
      HXS(K)=FLOFLD(1,K)
      HWS(K)=FLOFLD(2,K)
      RADS(K)=FLOFLD(3,K)
      XSURFS(K)=FLOFLD(4,K)
      US(K)=FLOFLD(5,K)
```

DATE 05/13/77

```
      VS(K)=FLOFLD(6,K)                                                 MODO  90
      X(K)=FLOFLD(7,K)                                                  MODO  91
      USX(K)=FLOFLD(8,K)                                                MODO  92
      VSX(K)=FLOFLD(9,K)                                                MODO  93
      RADSX(K)=FLOFLD(10,K)                                             MODO  94
      RADSXX(K)=FLOFLD(11,K)                                            MODO  95
      HXSX(K)=FLOFLD(12,K)                                              MODO  96
      HWSX(K)=FLOFLD(13,K)                                              MODO  97
      XX(K)=FLOFLD(14,K)                                                MODO  98
   70 CONTINUE                                                          MODO  99
      VS(1)=0.D0                                                        MODO 100
      VS(KL)=0.D0                                                       MODO 101
      VSX(1)=0.D0                                                       MODO 102
      VSX(KL)=0.D0                                                      MODO 103
C                                                                       MODO 104
C     EDGE PROPERTIES ARE CONVERTED TO FOURIER COEFFICIENTS             MODO 105
C                                                                       MODO 106
      IF (KL.EQ.1) GO TO 80                                             MODO 107
      CALL FORIER (RADS,ARADS,KL,1)                                     MODO 108
      CALL FORIER (US,AUS,KL,1)                                         MODO 109
      CALL FORIER (VS,AVS,KL,2)                                         MODO 110
      CALL FORIER (HXS,AHXS,KL,1)                                       MODO 111
      CALL FORIER (HWS,AHWS,KL,1)                                       MODO 112
      CALL FORIER (XSURFS,AXSRFS,KL,1)                                  MODO 113
      CALL FORIER (USX,AUSX,KL,1)                                       MODO 114
      CALL FORIER (VSX,AVSX,KL,2)                                       MODO 115
      CALL FORIER (RADSX,ARADSX,KL,1)                                   MODO 116
      CALL FORIER (RADSXX,ARDSXX,KL,1)                                  MODO 117
      CALL FORIER (HXSX,AHXSX,KL,1)                                     MODO 118
      CALL FORIER (HWSX,AHWSX,KL,1)                                     MODO 119
      CALL FORIER (X,AX,KL,1)                                           MODO 120
      CALL FORIER (XX,AXX,KL,1)                                         MODO 121
      GO TO 100                                                         MODO 122
   80 CONTINUE                                                          MODO 123
      ARADS(1)=RADS(1)                                                  MODO 124
      AUS(1)=US(1)                                                      MODO 125
      AVS(1)=VS(1)                                                      MODO 126
      AHXS(1)=HXS(1)                                                    MODO 127
      AHWS(1)=HWS(1)                                                    MODO 128
      AXSRFS(1)=XSURFS(1)                                               MODO 129
      AUSX(1)=USX(1)                                                    MODO 130
      AVSX(1)=VSX(1)                                                    MODO 131
      ARADSX(1)=RADSX(1)                                                MODO 132
      ARDSXX(1)=RADSXX(1)                                               MODO 133
      AHXSX(1)=HXSX(1)                                                  MODO 134
      AHWSX(1)=HWSX(1)                                                  MODO 135
      AX(1)=X(1)                                                        MODO 136
      AXX(1)=XX(1)                                                      MODO 137
      DO 90 I=2,KL                                                      MODO 138
      ARADS(I)=0.0D0                                                    MODO 139
      AUS(I)=0.D0                                                       MODO 140
      AVS(I)=0.0D0
      AHXS(I)=0.D0
      AHWS(I)=0.D0
      AXSRFS(I)=0.D0
      AUSX(I)=0.D0
      AVSX(I)=0.D0
      ARADSX(I)=0.D0
      ARDSXX(I)=0.D0
      AHXSX(I)=0.D0
      AHWSX(I)=0.D0
      AX(I)=0.D0
      AXX(I)=0.D0
90    CONTINUE
100   CONTINUE
      WRITE (30,210)
      WRITE (30,140)
      WRITE (10) XS,ARADS,AUS,AVS,AHXS,AHWS,AXSRFS,AUSX,AVSX,ARADSX,ARDS
     1XX,AHXSX,AHWSX,AX,AXX
      WRITE (30,220)
      DO 110 K=1,KL
      WRITE (30,200) K,AXSRFS(K),ARADS(K),AHXS(K),AHWS(K),AUS(K),AVS(K)
110   CONTINUE
      WRITE (30,230)
      DO 120 K=1,KL
      WRITE (30,200) K,AUSX(K),AVSX(K),ARADSX(K),ARDSXX(K),AHXSX(K),AHWS
     1X(K),AX(K),AXX(K)
120   CONTINUE
      WRITE (30,140)
      GO TO 30
C
C
130   RETURN
C
C
C
140   FORMAT (1H )
150   FORMAT (10X,39HNUMBER OF PLANES IN THE INVISCID DATA =,I3)
160   FORMAT (10X,6HALPHA=,F6.2)
170   FORMAT (5X,20HWALL DATA AT STATION,1X,I3,7H, XI = ,E9.4)
180   FORMAT (/5X,1HK,4X,4HHPSI,11X,2HHW,13X,1HR,14X,1HS,14X,4HUPSI,11X,
     12HUW,13X,1HX/)
190   FORMAT (/5X,1HK,4X,7HD(UPSI),8X,5HD(UW),10X,4HD(R),11X,5HD2(R),10X
     1,7HD(HPSI),8X,5HD(HW),10X,4HD(X)/)
200   FORMAT (4X,I2,8E15.6)
210   FORMAT (30X,20HFOURIER COEFFICIENTS)
220   FORMAT (/5X,1HK,4X,6HAXSRFS,9X,5HARADS,10X,4HAHXS,11X,4HAHWS,11X,3
     1HAUS,12X,3HAVS/)
230   FORMAT (/5X,1HK,4X,4HAUSX,11X,4HAVSX,11X,6HARADSX,9X,6HARDSXX,9X,5
     1HAHXSX,10X,5HAHWSX,10X,2HAX,13X,3HAXX/)
240   FORMAT (10X,58HUNIFORM FLOW STARTING SOLUTION FOR THE INVISCID FLO
     1W FIELD)
```

```
DATE  05/13/77

250    FORMAT (40X,52HINVISCID EDGE CONDITIONS FOR BOUNDARY LAYER SOLUTIOMODO 192
      1N/47X,39HTAKEN FROM THE INVISCID FLOW FIELD DATA)                 MODO 193
       END                                                               MODO 194


       SUBROUTINE FORIER (AR,BR,KL,KK)                                   MODO   1
C
C      SUBROUTINE FORIER FITS THE DATA IN ARRAY AR WITH EITHER A FOURIER
C      SINE OR COSINE SERIES.
C                                                                        MODO   2
C      BR IS THE OUTPUT ARRAY OF FOURIER COEFFICIENTS                    MODO   3
C      KL IS THE NUMBER OF INPUT DATA POINTS                             MODO   4
C      KK = 1 OUTPUTS COEFFICIENTS FOR A COSINE SERIES                   MODO   5
C      KK = 2 OUTPUTS COEFFICIENTS FOR A SINE SERIES                     MODO   6
C                                                                        MODO   7
C      AR IS THE INPUT ARRAY OF FUNCTION VALUES                          MODO   8
       IMPLICIT REAL*8(A-H,O-Z)                                          MODO   9
       DIMENSION AR(40), BR(40)                                          MODO  10
       PI=DARCOS(-1.0D0)                                                 MODO  11
       G=DFLOAT(KL-1)                                                    MODO  12
       FAC=2.0D0/G                                                       MODO  13
       IF (KK.EQ.2) GO TO 30                                             MODO  14
       M=KL-1                                                            MODO  15
       DO 20 N=1,KL                                                      MODO  16
       F=DFLOAT(N-1)                                                     MODO  17
       A=0.0D0                                                           MODO  18
       DO 10 K=2,M                                                       MODO  19
       E=DFLOAT(K-1)                                                     MODO  20
       A=A+AR(K)*DCOS(F*PI*E/G)                                          MODO  21
10     CONTINUE                                                          MODO  22
       BR(N)=FAC*(((AR(1)+AR(KL)*DCOS(F*PI))/2.0D0)+A)                   MODO  23
20     CONTINUE                                                          MODO  24
       BR(1)=BR(1)/2.0D0                                                 MODO  25
       BR(KL)=BR(KL)/2.0D0                                               MODO  26
       RETURN                                                            MODO  27
30     CONTINUE                                                          MODO  28
       DO 50 N=1,M                                                       MODO  29
       F=DFLOAT(N)                                                       MODO  30
       B=0.0D0                                                           MODO  31
       DO 40 K=2,M                                                       MODO  32
       E=DFLOAT(K-1)                                                     MODO  33
       B=B+AR(K)*DSIN(F*PI*E/G)                                          MODO  34
40     CONTINUE                                                          MODO  35
       BR(N)=FAC*B                                                       MODO  36
50     CONTINUE                                                          MODO  37
       RETURN                                                            MODO  38
       END                                                               MODO  39
```

APPENDIX VI

Job Control Language for Program ICBL3D

```
DATE  05/17/77


// JOB CARD
/*PRIORITY URGENT
/*JOBPARM LINES=10
//STEP1 EXEC FORTGCG,TIME=1,REGION=520K,PARM.FORT='NOSOURCE'
//FORT.SYSIN DD *

                     ICBL3D SOURCE DECK

//GO.SYSIN DD *

                     ICBL3D DATA DECK

//GO.FT06F001  DD SYSOUT=A
//GO.FT07F001  DD DUMMY
//GO.FT08F001  DD DSN=$$FT08,DISP=(NEW,DELETE),
//                UNIT=SYSDA,SPACE=(6472,(61,10),,CONTIG),
//                DCB=(RECFM=F,BLKSIZE=6472,DSORG=DA)
//GO.FT10F001  DD DSN=A505F3.INVTAP.EL412,DISP=(OLD,KEEP)
/*
//
```

APPENDIX VII

ICBL3D Sample Input

```
FILE: FSTR43E  DATA      A   VIRGINIA TECH CONVERSATIONAL MONITOR SYSTEM

        4:1 ELLIPSOID AT 2 DEG - - WITHOUT CURVATURE EFFECTS
      IE                                                61
   INJCT                                                 0
  KADETA                                                 1
   KEND2                                                10
  KONSET                                                 0
    KPRT                                                 1
  KTRANS
  LAMTRB                                                 1
    LPRT                                                 1
    NIT1                                                 7
    NIT2                                                10
    NIT3                                                15
   NDINJ
     SFC                                                NO
  NSOLVE                                                 4
  ADIFST                                                       .001
  AKSTAR                                                       .435
  ALAMDA                                                        .09
   ASTAR                                                        26.
   CWALL
    CONV                                                       .001
   DXMAX                                                        .10
     DX1                                                        .01
  EDYLAW                                               REI
  ETAFAC                                                       1.00
  ETAINF                                                         9.
      PR                                                         1.
     RTW                                        9.9983378E-01
     TFS                                                       500.
      CP                                                      6006.
  AMUINF                                                 7.650E-02
   PSTAG                                                   2116.80
    PINF                                                    1734.3
    XBAR                                                         1.
     UFS                                                       100.
       0.
   .0600
      .3
     1.5
```

APPENDIX VIII

ICBL3D Sample Output

```
                         THREE-DIMENSIONAL BOUNDARY LAYER PROGRAM
                                          FOR
                               LAMINAR OR TURBULENT FLOW
                                          WITH
                               SURFACE CURVATURE EFFECTS
                                      DEVELOPED BY
                             AEROSPACE ENGINEERING DEPARTMENT
                    VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY
                                 BLACKSBURG, VA. 24061


               4:1 ELLIPSOID AT 2 DEG - - WITHOUT CURVATURE EFFECTS

          FREE STREAM, STAGNATION, AND VEHICLE DATA:

          PSTAG = 0.211680D 04 PSIA
          TSTAG = 0.500833D 03 DEG.R
          PINF  = 0.173430D 04 PSIA
          UINF  = 0.100000D 03 FT/SEC
          TINF  = 0.500000D 03 DEG.R
          RHOINF= 0.765000D-01 SLUGS/FT**3
          AMUINF= 0.765000D-01 SLUGS/FT-SEC
          TW/TO = 0.999834D 00
          ALPHA = 0.200000D 01 DEG.
          CP    = 0.600600D 04 FT**2/(SEC**2-DEG.R)
          REINF = 0.100000D 03
          PR    = 0.100000D 01
          POINTS AT WHICH A SOLUTION IS TO BE OBTAINED:

            I      XSTA(I)

            1      0.0
            2      0.060000
            3      0.300000
            4      1.500000

XO    = 0.0               R     = 0.0             PHI   =  0.0  DEG.
XI    = 0.0               DXI   = 0.0             CWALL= 0.0              NIT  =  4
HX    = 0.100000D 01      HW    = 0.0

NONDIMENSIONAL EDGE PROPERTIES

TE    = 0.300300D 03      UE    = 0.0              VF    = 0.0
DUEDX= 0.428070D 01       DVEDX= 0.0               DUEDW= 0.0              DVEDW= 0.0

LOCAL EDGE REYNOLDS NUMBER =0.0

NONDIMENSIONAL BOUNDARY LAYER PARAMETERS

CFXINF= 0.0               CFXEDG= 0.0            CFWINF= 0.0             CFWEDG= 0.0
QW     =-0.787501D-02     CHIMAX= 0.0

DIMENSIONAL BOUNDARY LAYER PARAMETERS

LONGITUDINAL SKIN FRICTION= 0.0         PSF      DELTA*(X)   = 0.275387D-01     THETA(X)   = 0.119006D-01
TRANSVERSE SKIN FRICTION  = 0.0         PSF      DELTA*(PHI)= 0.305399D-01      THETA(PHI)= 0.131764D-01
```

```
   WALL HEAT TRANSFER RATE   =-0.774735D 00 BTU     DELTA (FT) = 0.102466D 00

  ETA            Y           F            FN          G           GN           T           TN           V           EPLUS

0.0          0.0         0.0          0.927864    0.0         0.618048    0.999834    0.000090   0.0            0.0
0.300000     0.103D-01   0.255876     0.778115    0.184767    0.610967    0.999861    0.000089  -0.392D-01      0.0
0.600000     0.205D-01   0.467015     0.630470    0.364616    0.530028    0.999887    0.000087  -0.148D 00      0.0
0.900000     0.308D-01   0.634652     0.489511    0.530279    0.516594    0.999912    0.000081  -0.315D 00      0.0
1.200000     0.410D-01   0.761682     0.361340    0.672608    0.427451    0.999935    0.000072  -0.525D 00      0.0
1.500000     0.513D-01   0.852861     0.251774    0.785901    0.327040    0.999955    0.000059  -0.786D 00      0.0
1.800000     0.615D-01   0.914439     0.164568    0.869161    0.230744    0.999971    0.000045  -0.103D 01      0.0
2.100000     0.718D-01   0.953344     0.100386    0.925534    0.149865    0.999982    0.000032  -0.131D 01      0.0
2.400000     0.820D-01   0.976232     0.056908    0.960645    0.089479    0.999990    0.000021  -0.160D 01      0.0
2.700000     0.923D-01   0.988725     0.029883    0.980735    0.049056    0.999995    0.000012  -0.190D 01      0.0
3.000000     0.103D 00   0.995032     0.014498    0.991285    0.024669    0.999997    0.000007  -0.220D 01      0.0
3.300000     0.113D 00   0.997971     0.006486    0.996364    0.011367    0.999999    0.000003  -0.249D 01      0.0
3.600000     0.123D 00   0.999233     0.002672    0.998604    0.004795    1.000000    0.000001  -0.279D 01      0.0
3.900000     0.133D 00   0.999731     0.001013    0.999507    0.001849    1.000000    0.000001  -0.309D 01      0.0
4.200000     0.144D 00   0.999913     0.000354    0.999840    0.000652    1.000000    0.000000  -0.339D 01      0.0
4.500000     0.154D 00   0.999973     0.000114    0.999953    0.000210    1.000000    0.000000  -0.369D 01      0.0
4.800000     0.164D 00   0.999992     0.000034    0.999987    0.000062    1.000000    0.000000  -0.399D 01      0.0
5.100000     0.174D 00   0.999998     0.000010    0.999997    0.000017    1.000000    0.000000  -0.429D 01      0.0
5.400000     0.185D 00   0.999999     0.000003    0.999999    0.000004    1.000000    0.000000  -0.459D 01      0.0
5.700000     0.195D 00   1.000000     0.000001    1.000000    0.000001    1.000000    0.000000  -0.489D 01      0.0
6.000000     0.205D 00   1.000000     0.000000    1.000000    0.000000    1.000000    0.000000  -0.519D 01      0.0
6.300000     0.215D 00   1.000000     0.000000    1.000000    0.000000    1.000000    0.000000  -0.549D 01      0.0
6.600000     0.226D 00   1.000000     0.000000    1.000000    0.000000    1.000000    0.000000  -0.579D 01      0.0
6.900000     0.236D 00   1.000000     0.000000    1.000000    0.000000    1.000000    0.000000  -0.609D 01      0.0
7.200000     0.246D 00   1.000000     0.000000    1.000000    0.000000    1.000000    0.000000  -0.639D 01      0.0
7.500000     0.256D 00   1.000000     0.000000    1.000000    0.000000    1.000000    0.000000  -0.669D 01      0.0
7.800000     0.267D 00   1.000000     0.000000    1.000000    0.000000    1.000000    0.000000  -0.699D 01      0.0
8.100000     0.277D 00   1.000000     0.000000    1.000000    0.000000    1.000000    0.000000  -0.729D 01      0.0
8.400000     0.287D 00   1.000000     0.000000    1.000000    0.000000    1.000000    0.000000  -0.759D 01      0.0
8.700000     0.297D 00   1.000000     0.000000    1.000000    0.000000    1.000000    0.000000  -0.789D 01      0.0
9.000000     0.308D 00   1.000000     0.000000    1.000000    0.000000    1.000000    0.0       -0.819D 01      0.0

                                                 *****          *****          *****

   XO    = 0.0              R     = 0.0              PHI  = 20.00 DEG.
   XI    = 0.0              DXI   = 0.0              CWALL= 0.0               NIT  =  3
   HX    = 0.100000D 01     HW    = 0.0

   NONDIMENSIONAL EDGE PROPERTIES

   TE    = 0.300300D 03     UE    = 0.0              VE    = 0.0
   DUEDX= 0.440919D 01      DVEDX=-0.268870D-01      DUEDW= 0.0                    DVEDW= 0.0

   LOCAL EDGE REYNOLDS NUMBER =0.0

   NONDIMENSIONAL BOUNDARY LAYER PARAMETERS

   CFXINF= 0.0              CFXEDG= 0.0              CFWINF= 0.0                   CFWEDG= 0.0
   QW     =-0.811431D-02    CHIMAX= 0.0

   DIMENSIONAL BOUNDARY LAYER PARAMETERS

   LONGITUDINAL SKIN FRICTION= 0.0          PSF     DELTA*(X)   = 0.269183D-01     THETA(X)   = 0.116160D-01
   TRANSVERSE SKIN FRICTION  = 0.0          PSF     DELTA*(PHI)= 0.192940D-01      THETA(PHI)= 0.100709D-01
   WALL HEAT TRANSFER RATE   =-0.798278D 00 BTU     DELTA (FT) = 0.102772D 00

  ETA            Y           F            FN          G           GN           T           TN           V           EPLUS
```

```
0.0         0.0         0.0         0.932627    0.0         1.259941    0.999834    0.000091  0.0         0.0
0.300000    0.101D-01   0.257247    0.782398    0.333017    0.962803    0.999861    0.000090 -0.427D-01   0.0
0.600000    0.202D-01   0.469529    0.633644    0.579691    0.690952    0.999888    0.000088 -0.157D 00   0.0
0.900000    0.303D-01   0.637891    0.491094    0.750952    0.463335    0.999914    0.000082 -0.329D 00   0.0
1.200000    0.404D-01   0.765140    0.361259    0.861671    0.283207    0.999937    0.000072 -0.547D 00   0.0
1.500000    0.505D-01   0.856078    0.250367    0.927792    0.164859    0.999956    0.000059 -0.797D 00   0.0
1.800000    0.606D-01   0.917098    0.162423    0.963928    0.085840    0.999972    0.000045 -0.107D 01   0.0
2.100000    0.707D-01   0.955319    0.098108    0.981800    0.040198    0.999983    0.000031 -0.136D 01   0.0
2.400000    0.808D-01   0.977558    0.054934    0.989698    0.016716    0.999991    0.000020 -0.165D 01   0.0
2.700000    0.909D-01   0.989531    0.028415    0.992785    0.006157    0.999995    0.000011 -0.196D 01   0.0
3.000000    0.101D 00   0.995477    0.013537    0.993872    0.002155    0.999998    0.000006 -0.226D 01   0.0
3.300000    0.111D 00   0.998193    0.005926    0.994281    0.000985    0.999999    0.000003 -0.257D 01   0.0
3.600000    0.121D 00   0.999331    0.002379    0.994532    0.000809    1.000000    0.000001 -0.287D 01   0.0
3.900000    0.131D 00   0.999769    0.000874    0.994782    0.000877    1.000000    0.000001 -0.318D 01   0.0
4.200000    0.141D 00   0.999923    0.000294    0.995058    0.000954    1.000000    0.000000 -0.349D 01   0.0
4.500000    0.152D 00   0.999972    0.000091    0.995352    0.000998    1.000000    0.000000 -0.379D 01   0.0
4.800000    0.162D 00   0.999987    0.000026    0.995655    0.001018    1.000000    0.000000 -0.410D 01   0.0
5.100000    0.172D 00   0.999991    0.000008    0.995962    0.001026    1.000000    0.000000 -0.441D 01   0.0
5.400000    0.182D 00   0.999992    0.000003    0.996271    0.001029    1.000000    0.000000 -0.471D 01   0.0
5.700000    0.192D 00   0.999993    0.000002    0.996580    0.001031    1.000000    0.000000 -0.502D 01   0.0
6.000000    0.202D 00   0.999994    0.000002    0.996889    0.001032    1.000000    0.000000 -0.533D 01   0.0
6.300000    0.212D 00   0.999994    0.000002    0.997199    0.001033    1.000000    0.000000 -0.563D 01   0.0
6.600000    0.222D 00   0.999995    0.000002    0.997509    0.001034    1.000000    0.000000 -0.594D 01   0.0
6.900000    0.232D 00   0.999995    0.000002    0.997820    0.001035    1.000000    0.000000 -0.625D 01   0.0
7.200000    0.242D 00   0.999996    0.000002    0.998130    0.001036    1.000000    0.000000 -0.655D 01   0.0
7.500000    0.253D 00   0.999997    0.000002    0.998441    0.001037    1.000000    0.000000 -0.686D 01   0.0
7.800000    0.263D 00   0.999997    0.000002    0.998753    0.001038    1.000000    0.000000 -0.717D 01   0.0
8.100000    0.273D 00   0.999998    0.000002    0.999064    0.001039    1.000000    0.0      -0.748D 01   0.0
8.400000    0.283D 00   0.999999    0.000002    0.999376    0.001040    1.000000    0.0      -0.778D 01   0.0
8.700000    0.293D 00   0.999999    0.000002    0.999688    0.001040    1.000000    0.000000 -0.809D 01   0.0
9.000000    0.303D 00   1.000000    0.000002    1.000000    0.001041    1.000000    0.000000 -0.840D 01   0.0
                                                 *****       *****       *****

      XO   = 0.0                R    = 0.0               PHI   = 40.00 DEG.
      XI   = 0.0                DXI  = 0.0               CWALL= 0.0               NIT  =  2
      HX   = 0.1000000D 01      HW   = 0.0

      NONDIMENSIONAL EDGE PROPERTIES

      TE    = 0.3003000D 03     UE    = 0.0              VE    = 0.0
      DUEDX= 0.4537960D 01      DVEDX=-0.7482320D-01     DUEDW= 0.0               DVEDW= 0.0

      LOCAL EDGE REYNOLDS NUMBER =0.0

      NONDIMENSIONAL BOUNDARY LAYER PARAMETERS

      CFXINF= 0.0               CFXEDG= 0.0              CFWINF= 0.0              CFWEDG= 0.0
      QW     =-0.8222300D-02    CHIMAX= 0.0

      DIMENSIONAL BOUNDARY LAYER PARAMETERS

      LONGITUDINAL SKIN FRICTION= 0.0            PSF    DELTA*(X)   = 0.265519D-01     THETA(X)   = 0.114586D-01
      TRANSVERSE SKIN FRICTION   = 0.0           PSF    DELTA*(PHI)= 0.178395D-01      THETA(PHI)= 0.888894D-02
      WALL HEAT TRANSFER RATE    =-0.808901D 00 BTU     DELTA (FT) = 0.911607D-01

   ETA          Y           F           FN          G           GN          T           TN          V          EPLUS

0.0         0.0         0.0         0.931822    0.0         1.269805    0.999834    0.000091  0.0         0.0
0.300000    0.996D-02   0.257041    0.781866    0.335589    0.970157    0.999861    0.000090 -0.407D-01   0.0
0.600000    0.199D-01   0.469226    0.633538    0.584139    0.696195    0.999888    0.000088 -0.154D 00   0.0
0.900000    0.299D-01   0.637613    0.491320    0.756691    0.466760    0.999913    0.000082 -0.327D 00   0.0
1.200000    0.398D-01   0.764957    0.361610    0.868197    0.290107    0.999937    0.000072 -0.545D 00   0.0
```

```
1.500000   0.498D-01   0.855997   0.250666   0.934698   0.165590   0.999956   0.000059 -0.797D 00   0.0
1.800000   0.597D-01   0.917088   0.162584   0.970920   0.085774   0.999972   0.000045 -0.107D 01   0.0
2.100000   0.697D-01   0.955335   0.098135   0.988691   0.039655   0.999983   0.000031 -0.136D 01   0.0
2.400000   0.797D-01   0.977568   0.054880   0.996382   0.015921   0.999991   0.000020 -0.166D 01   0.0
2.700000   0.896D-01   0.989519   0.023335   0.999209   0.005245   0.999995   0.000011 -0.196D 01   0.0
3.000000   0.996D-01   0.995442   0.013470   1.000013   0.001191   0.999998   0.000006 -0.227D 01   0.0
3.300000   0.110D 00   0.998141   0.005883   1.000129  -0.000002   0.999999   0.000003 -0.258D 01   0.0
3.600000   0.119D 00   0.999271   0.002359   1.000082  -0.000191   1.000000   0.000001 -0.289D 01   0.0
3.900000   0.129D 00   0.999705   0.000871   1.000031  -0.000131   1.000000   0.000001 -0.320D 01   0.0
4.200000   0.139D 00   0.999859   0.000300   1.000003  -0.000059   1.000000   0.000000 -0.350D 01   0.0
4.500000   0.149D 00   0.999911   0.000101   0.999993  -0.000019   1.000000   0.000000 -0.381D 01   0.0
4.800000   0.159D 00   0.999929   0.000039   0.999990  -0.000002   1.000000   0.000000 -0.412D 01   0.0
5.100000   0.169D 00   0.999937   0.000021   0.999990   0.000003   1.000000   0.000000 -0.443D 01   0.0
5.400000   0.179D 00   0.999943   0.000017   0.999992   0.000004   1.000000   0.000000 -0.474D 01   0.0
5.700000   0.189D 00   0.999948   0.000016   0.999993   0.000004   1.000000   0.000000 -0.505D 01   0.0
6.000000   0.199D 00   0.999952   0.000016   0.999994   0.000004   1.000000   0.000000 -0.536D 01   0.0
6.300000   0.209D 00   0.999957   0.000016   0.999995   0.000003   1.000000   0.000000 -0.566D 01   0.0
6.600000   0.219D 00   0.999962   0.000016   0.999996   0.000003   1.000000   0.000000 -0.597D 01   0.0
6.900000   0.229D 00   0.999966   0.000016   0.999997   0.000003   1.000000   0.000000 -0.628D 01   0.0
7.200000   0.239D 00   0.999971   0.000016   0.999998   0.000002   1.000000   0.000000 -0.659D 01   0.0
7.500000   0.249D 00   0.999976   0.000016   0.999999   0.000002   1.000000   0.000000 -0.690D 01   0.0
7.800000   0.259D 00   0.999981   0.000016   0.999999   0.000001   1.000000   0.0      -0.721D 01   0.0
8.100000   0.269D 00   0.999986   0.000016   0.999999   0.000001   1.000000   0.0      -0.752D 01   0.0
8.400000   0.279D 00   0.999990   0.000016   1.000000   0.000001   1.000000  -0.000000 -0.783D 01   0.0
8.700000   0.289D 00   0.999995   0.000016   1.000000   0.000000   1.000000   0.000000 -0.813D 01   0.0
9.000000   0.299D 00   1.000000   0.000016   1.000000   0.000000   1.000000   0.000000 -0.844D 01   0.0

                                            *****          *****          *****

       XO    = 0.0              R     = 0.0              PHI   = 60.00 DEG.
       XI    = 0.0              DXI   = 0.0              CWALL= 0.0               NIT   =  2
       HX    = 0.100000D 01     HW    = 0.0

       NONDIMENSIONAL EDGE PROPERTIES

       TE    = 0.300300D 03     UE    = 0.0              VE    = 0.0
       DUEDX= 0.475684D 01      DVEDX=-0.166012D 00      DUEDW= 0.0                      DVEDW= 0.0

       LOCAL EDGE REYNOLDS NUMBER =0.0

       NONDIMENSIONAL BOUNDARY LAYER PARAMETERS

       CFXINF= 0.0              CFXEDG= 0.0                  CFWINF= 0.0                CFWEDG= 0.0
       QW     =-0.850448D-02    CHIMAX= 0.0

       DIMENSIONAL BOUNDARY LAYER PARAMETERS

       LONGITUDINAL SKIN FRICTION= 0.0           PSF    DELTA*(X)   = 0.258372D-01    THETA(X)   = 0.111606D-01
       TRANSVERSE SKIN FRICTION  = 0.0           PSF    DELTA*(PHI)= 0.172446D-01     THETA(PHI)= 0.858594D-02
       WALL HEAT TRANSFER RATE     =-0.836662D 00 BTU   DELTA (FT) = 0.883637D-01

   ETA          Y           F          FN         G          GN          T          TN          V          EPLUS

0.0        0.0         0.0         0.934920   0.0        1.279138   0.999834   0.000092  0.0          0.0
0.300000   0.973D-02   0.257923   0.784612   0.337768   0.975409   0.999861   0.000091 -0.418D-01   0.0
0.600000   0.195D-01   0.470848   0.635624   0.587337   0.697945   0.999888   0.000089 -0.159D 00   0.0
0.900000   0.292D-01   0.639718   0.492387   0.759996   0.466004   0.999914   0.000082 -0.337D 00   0.0
1.200000   0.389D-01   0.767208   0.361523   0.871028   0.287996   0.999938   0.000072 -0.561D 00   0.0
1.500000   0.486D-01   0.858064   0.249608   0.936813   0.163144   0.999957   0.000059 -0.820D 00   0.0
1.800000   0.584D-01   0.918736   0.160968   0.972335   0.083669   0.999973   0.000044 -0.110D 01   0.0
2.100000   0.681D-01   0.956468   0.096422   0.989564   0.038173   0.999984   0.000030 -0.140D 01   0.0
2.400000   0.778D-01   0.978215   0.053414   0.996905   0.015039   0.999991   0.000019 -0.170D 01   0.0
2.700000   0.875D-01   0.989784   0.027273   0.999538   0.004791   0.999996   0.000011 -0.202D 01   0.0
```

```
3.000000    0.973D-01   0.995450   0.012807   1.000249    0.000981   0.999998   0.000005 -0.233D 01   0.0
3.300000    0.107D 00   0.998001   0.005530   1.000323   -0.000103   0.999999   0.000003 -0.264D 01   0.0
3.600000    0.117D 00   0.999059   0.002205   1.000252   -0.000255   1.000000   0.000001 -0.296D 01   0.0
3.900000    0.126D 00   0.999466   0.000828   1.000184   -0.000188   1.000000   0.000000 -0.327D 01   0.0
4.200000    0.136D 00   0.999617   0.000312   1.000139   -0.000115   1.000000   0.000000 -0.359D 01   0.0
4.500000    0.146D 00   0.999677   0.000137   1.000112   -0.000074   1.000000   0.000000 -0.391D 01   0.0
4.800000    0.156D 00   0.999708   0.000084   1.000093   -0.000055   1.000000   0.000000 -0.422D 01   0.0
5.100000    0.165D 00   0.999730   0.000070   1.000079   -0.000045   1.000000   0.000000 -0.454D 01   0.0
5.400000    0.175D 00   0.999751   0.000067   1.000066   -0.000039   1.000000   0.000000 -0.485D 01   0.0
5.700000    0.185D 00   0.999771   0.000067   1.000055   -0.000035   1.000000   0.000000 -0.517D 01   0.0
6.000000    0.195D 00   0.999791   0.000068   1.000045   -0.000031   1.000000   0.000000 -0.548D 01   0.0
6.300000    0.204D 00   0.999812   0.000068   1.000036   -0.000027   1.000000   0.000000 -0.580D 01   0.0
6.600000    0.214D 00   0.999832   0.000069   1.000029   -0.000024   1.000000   0.000000 -0.611D 01   0.0
6.900000    0.224D 00   0.999853   0.000069   1.000022   -0.000021   1.000000   0.000000 -0.643D 01   0.0
7.200000    0.233D 00   0.999874   0.000069   1.000016   -0.000018   1.000000   0.000000 -0.674D 01   0.0
7.500000    0.243D 00   0.999895   0.000070   1.000011   -0.000015   1.000000   0.000000 -0.706D 01   0.0
7.800000    0.253D 00   0.999916   0.000070   1.000007   -0.000012   1.000000   0.000000 -0.738D 01   0.0
8.100000    0.263D 00   0.999937   0.000070   1.000004   -0.000009   1.000000  -0.000000 -0.769D 01   0.0
8.400000    0.272D 00   0.999958   0.000070   1.000002   -0.000006   1.000000   0.0      -0.801D 01   0.0
8.700000    0.282D 00   0.999979   0.000070   1.000001   -0.000003   1.000000   0.000000 -0.832D 01   0.0
9.000000    0.292D 00   1.000000   0.000070   1.000000   -0.000001   1.000000  -0.000000 -0.864D 01   0.0

                                         *****            *****            *****

      XO    = 0.0                 R     = 0.0              PHI   = 80.00 DEG.
      XI    = 0.0                 DXI   = 0.0              CWALL= 0.0                NIT   =  2
      HX    = 0.100000D 01        HW    = 0.0

      NONDIMENSIONAL EDGE PROPERTIES

      TE    = 0.300300D 03        UE    = 0.0              VE    = 0.0
      DUEDX= 0.505230D 01         DVEDX=-0.299270D 00      DUEDW= 0.0                          DVEDW= 0.0

      LOCAL EDGE REYNOLDS NUMBER =0.0

      NONDIMENSIONAL BOUNDARY LAYER PARAMETERS

      CFXINF= 0.0                 CFXEDG= 0.0                    CFWINF= 0.0                  CFWEDG= 0.0
      QW    =-0.882051D-02        CHIMAX= 0.0

      DIMENSIONAL BOUNDARY LAYER PARAMETERS

      LONGITUDINAL SKIN FRICTION= 0.0           PSF      DELTA*(X)   = 0.250883D-01      THETA(X)   = 0.108855D-01
      TRANSVERSE SKIN FRICTION  = 0.0           PSF      DELTA*(PHI)= 0.166339D-01      THETA(PHI)= 0.826343D-02
      WALL HEAT TRANSFER RATE   =-0.867753D 00 BTU      DELTA (FT)  = 0.857233D-01

  ETA            Y            F            FN           G            GN           T            TN          V            EPLUS

0.0          0.0          0.0          0.937403    0.0          1.282273    0.999834    0.000092  0.0           0.0
0.300000     0.944D-02    0.258568     0.786421    0.338540     0.977427    0.999861    0.000092 -0.424D-01     0.0
0.600000     0.189D-01    0.471928     0.636690    0.588551     0.698898    0.999889    0.000089 -0.162D 00     0.0
0.900000     0.283D-01    0.640996     0.492638    0.761352     0.466055    0.999915    0.000083 -0.343D 00     0.0
1.200000     0.378D-01    0.768442     0.361012    0.872297     0.287452    0.999938    0.000072 -0.571D 00     0.0
1.500000     0.472D-01    0.859051     0.248544    0.937870     0.162358    0.999958    0.000059 -0.834D 00     0.0
1.800000     0.566D-01    0.919355     0.159661    0.973154     0.082924    0.999973    0.000044 -0.112D 01     0.0
2.100000     0.661D-01    0.956694     0.095170    0.990184     0.037611    0.999984    0.000030 -0.142D 01     0.0
2.400000     0.755D-01    0.978098     0.052416    0.997387     0.014675    0.999991    0.000019 -0.173D 01     0.0
2.700000     0.849D-01    0.989416     0.026601    0.999937     0.004573    0.999996    0.000011 -0.205D 01     0.0
3.000000     0.944D-01    0.994928     0.012431    1.000596     0.000840    0.999998    0.000005 -0.237D 01     0.0
3.300000     0.104D 00    0.997402     0.005372    1.000633    -0.000213    0.999999    0.000003 -0.268D 01     0.0
3.600000     0.113D 00    0.998435     0.002185    1.000531    -0.000357    1.000000    0.000001 -0.300D 01     0.0
3.900000     0.123D 00    0.998850     0.000883    1.000433    -0.000287    1.000000    0.000000 -0.332D 01     0.0
4.200000     0.132D 00    0.999024     0.000404    1.000359    -0.000212    1.000000    0.000000 -0.364D 01     0.0
```

```
4.500000    0.142D 00    0.999115    0.000246    1.000303   -0.000165    1.000000    0.000000 -0.396D 01   0.0
4.800000    0.151D 00    0.999180    0.000200    1.000258   -0.000139    1.000000    0.000000 -0.428D 01   0.0
5.100000    0.160D 00    0.999238    0.000190    1.000219   -0.000122    1.000000    0.000000 -0.460D 01   0.0
5.400000    0.170D 00    0.999294    0.000189    1.000184   -0.000109    1.000000    0.000000 -0.492D 01   0.0
5.700000    0.179D 00    0.999351    0.000191    1.000153   -0.000098    1.000000    0.000000 -0.524D 01   0.0
6.000000    0.189D 00    0.999409    0.000192    1.000126   -0.000087    1.000000    0.000000 -0.556D 01   0.0
6.300000    0.198D 00    0.999467    0.000194    1.000101   -0.000077    1.000000    0.000000 -0.588D 01   0.0
6.600000    0.208D 00    0.999525    0.000195    1.000080   -0.000067    1.000000    0.000000 -0.620D 01   0.0
6.900000    0.217D 00    0.999584    0.000196    1.000061   -0.000058    1.000000    0.000000 -0.652D 01   0.0
7.200000    0.227D 00    0.999642    0.000197    1.000045   -0.000049    1.000000    0.000000 -0.684D 01   0.0
7.500000    0.236D 00    0.999702    0.000198    1.000032   -0.000041    1.000000    0.000000 -0.716D 01   0.0
7.800000    0.245D 00    0.999761    0.000198    1.000021   -0.000032    1.000000    0.000000 -0.748D 01   0.0
8.100000    0.255D 00    0.999820    0.000199    1.000012   -0.000025    1.000000    0.000000 -0.780D 01   0.0
8.400000    0.264D 00    0.999880    0.000199    1.000006   -0.000017    1.000000    0.000000 -0.812D 01   0.0
8.700000    0.274D 00    0.999940    0.000200    1.000002   -0.000010    1.000000    0.000000 -0.844D 01   0.0
9.000000    0.283D 00    1.000000    0.000200    1.000000   -0.000003    1.000000    0.0        -0.876D 01   0.0

                                                  *****           *****           *****

     X0    = 0.0                R     = 0.0                 PHI   =100.00 DEG.
     XI    = 0.0                DXI   = 0.0                 CWALL= 0.0                NIT  =  2
     HX    = 0.1000000D 01      HW    = 0.0

     NONDIMENSIONAL EDGE PROPERTIES

     TE    = 0.3003000D 03      UE    = 0.0                 VE    = 0.0
     DUEDX= 0.5267390D 01       DVEDX=-0.4019200D 00        DUEDW= 0.0                        DVEDW= 0.0

     LOCAL EDGE REYNOLDS NUMBER =0.0

     NONDIMENSIONAL BOUNDARY LAYER PARAMETERS

     CFXINF= 0.0                CFXEDG= 0.0                 CFWINF= 0.0                       CFWEDG= 0.0
     QW     =-0.892456D-02      CHIMAX= 0.0

     DIMENSIONAL BOUNDARY LAYER PARAMETERS

     LONGITUDINAL SKIN FRICTION= 0.0          PSF     DELTA*(X)   = 0.248026D-01     THETA(X)   = 0.108330D-01
     TRANSVERSE SKIN FRICTION  = 0.0          PSF     DELTA*(PHI)= 0.164713D-01      THETA(PHI)= 0.814739D-02
     WALL HEAT TRANSFER RATE   =-0.877989D 00 BTU     DELTA (FT)  = 0.850817D-01

  ETA           Y            F            FN          G            GN          T            TN          V            EPLUS

0.0         0.0          0.0          0.935520    0.0          1.265914    0.999834    0.000092  0.0            0.0
0.300000    0.924D-02    0.257917     0.784030    0.334848     0.959015    0.999861    0.000091 -0.410D-01     0.0
0.600000    0.185D-01    0.470537     0.634314    0.583387     0.696979    0.999888    0.000089 -0.157D 00     0.0
0.900000    0.277D-01    0.638968     0.490899    0.756334     0.468353    0.999914    0.000082 -0.333D 00     0.0
1.200000    0.370D-01    0.766036     0.360271    0.868337     0.291678    0.999937    0.000072 -0.556D 00     0.0
1.500000    0.462D-01    0.856578     0.248791    0.935255     0.166745    0.999957    0.000059 -0.812D 00     0.0
1.800000    0.555D-01    0.917073     0.160592    0.971750     0.086453    0.999973    0.000044 -0.109D 01     0.0
2.100000    0.647D-01    0.954748     0.096378    0.989662     0.039954    0.999984    0.000031 -0.139D 01     0.0
2.400000    0.739D-01    0.976515     0.053566    0.997401     0.015964    0.999991    0.000019 -0.169D 01     0.0
2.700000    0.832D-01    0.988146     0.027513    1.000213     0.005127    0.999995    0.000011 -0.200D 01     0.0
3.000000    0.924D-01    0.993890     0.013072    1.000965     0.000976    0.999998    0.000006 -0.231D 01     0.0
3.300000    0.102D 00    0.996520     0.005792    1.001008    -0.000270    0.999999    0.000003 -0.262D 01     0.0
3.600000    0.111D 00    0.997655     0.002461    1.000876    -0.000482    1.000000    0.000001 -0.294D 01     0.0
3.900000    0.120D 00    0.998139     0.001079    1.000736    -0.000424    1.000000    0.000000 -0.325D 01     0.0
4.200000    0.129D 00    0.998365     0.000561    1.000622    -0.000340    1.000000    0.000000 -0.356D 01     0.0
4.500000    0.139D 00    0.998500     0.000387    1.000530    -0.000281    1.000000    0.000000 -0.383D 01     0.0
4.800000    0.148D 00    0.998606     0.000337    1.000452    -0.000243    1.000000    0.000000 -0.419D 01     0.0
5.100000    0.157D 00    0.998704     0.000325    1.000384    -0.000215    1.000000    0.000000 -0.450D 01     0.0
5.400000    0.166D 00    0.998801     0.000324    1.000323    -0.000192    1.000000    0.000000 -0.482D 01     0.0
5.700000    0.176D 00    0.998899     0.000326    1.000268    -0.000172    1.000000    0.000000 -0.513D 01     0.0
```

```
6.000000   0.185D 00   0.998997   0.000328   1.000220  -0.000152   1.000000   0.000000 -0.544D 01   0.0
6.300000   0.194D 00   0.999095   0.000329   1.000177  -0.000134   1.000000   0.000000 -0.576D 01   0.0
6.600000   0.203D 00   0.999194   0.000331   1.000139  -0.000117   1.000000   0.000000 -0.607D 01   0.0
6.900000   0.213D 00   0.999294   0.000333   1.000106  -0.000101   1.000000   0.000000 -0.639D 01   0.0
7.200000   0.222D 00   0.999394   0.000334   1.000078  -0.000086   1.000000   0.000000 -0.670D 01   0.0
7.500000   0.231D 00   0.999494   0.000335   1.000055  -0.000071   1.000000   0.000000 -0.701D 01   0.0
7.800000   0.240D 00   0.999595   0.000336   1.000036  -0.000056   1.000000   0.000000 -0.733D 01   0.0
8.100000   0.250D 00   0.999696   0.000337   1.000021  -0.000043   1.000000   0.000000 -0.764D 01   0.0
8.400000   0.259D 00   0.999797   0.000338   1.000010  -0.000030   1.000000   0.000000 -0.795D 01   0.0
8.700000   0.268D 00   0.999898   0.000338   1.000003  -0.000017   1.000000   0.0        -0.827D 01   0.0
9.000000   0.277D 00   1.000000   0.000339   1.000000  -0.000005   1.000000   0.0        -0.858D 01   0.0

                                         *****          *****          *****

     XU    = 0.0               R     = 0.0               PHI   =120.00 DEG.
     XI    = 0.0               DXI   = 0.0               CWALL= 0.0               NIT   =  2
     HX    = 0.100000D 01      HW    = 0.0

     NONDIMENSIONAL EDGE PROPERTIES

     TE    = 0.300300D 03      UE    = 0.0               VE    = 0.0
     DUEDX= 0.513851D 01       DVEDX=-0.338351D 00       DUEDW= 0.0               DVEDW= 0.0

     LOCAL EDGE REYNOLDS NUMBER =0.0

     NONDIMENSIONAL BOUNDARY LAYER PARAMETERS

     CFXINF= 0.0               CFXEDG= 0.0               CFWINF= 0.0               CFWEDG= 0.0
     QW     =-0.854976D-02     CHIMAX= 0.0

     DIMENSIONAL BOUNDARY LAYER PARAMETERS

     LONGITUDINAL SKIN FRICTION= 0.0          PSF     DELTA*(X)   = 0.254735D-01     THETA(X)   = 0.111394D-01
     TRANSVERSE SKIN FRICTION  = 0.0          PSF     DELTA*(PHI)= 0.172907D-01     THETA(PHI)= 0.853044D-02
     WALL HEAT TRANSFER RATE    =-0.841117D 00 BTU    DELTA (FT)  = 0.879576D-01

   ETA            Y            F            FN          G           GN          T            TN          V            EPLUS

0.0          0.0          0.0          0.926719   0.0          1.227126   0.999834   0.000089  0.0           0.0
0.300000     0.936D-02    0.255369     0.775962   0.325890     0.947782   0.999860   0.000088 -0.374D-01    0.0
0.600000     0.187D-01    0.465778     0.627951   0.570424     0.690524   0.999887   0.000086 -0.143D 00    0.0
0.900000     0.281D-01    0.632690     0.487359   0.743155     0.472120   0.999912   0.000081 -0.305D 00    0.0
1.200000     0.374D-01    0.759189     0.360023   0.857260     0.300750   0.999935   0.000071 -0.509D 00    0.0
1.500000     0.468D-01    0.850117     0.251395   0.927205     0.176989   0.999954   0.000059 -0.746D 00    0.0
1.800000     0.561D-01    0.911703     0.164932   0.966621     0.095218   0.999970   0.000045 -0.101D 01    0.0
2.100000     0.655D-01    0.950791     0.101161   0.986793     0.046156   0.999982   0.000032 -0.128D 01    0.0
2.400000     0.749D-01    0.973938     0.057789   0.996001     0.019680   0.999990   0.000021 -0.156D 01    0.0
2.700000     0.842D-01    0.986687     0.030678   0.999625     0.007010   0.999994   0.000013 -0.185D 01    0.0
3.000000     0.936D-01    0.993210     0.015137   1.000752     0.001764   0.999997   0.000007 -0.214D 01    0.0
3.300000     0.103D 00    0.996318     0.006979   1.000934    -0.000019   0.999999   0.000003 -0.244D 01    0.0
3.600000     0.112D 00    0.997710     0.003061   1.000838    -0.000438   1.000000   0.000002 -0.273D 01    0.0
3.900000     0.122D 00    0.998315     0.001342   1.000701    -0.000433   1.000000   0.000001 -0.302D 01    0.0
4.200000     0.131D 00    0.998590     0.000653   1.000583    -0.000350   1.000000   0.000000 -0.332D 01    0.0
4.500000     0.140D 00    0.998738     0.000400   1.000489    -0.000281   1.000000   0.000000 -0.361D 01    0.0
4.800000     0.150D 00    0.998842     0.000315   1.000412    -0.000235   1.000000   0.000000 -0.390D 01    0.0
5.100000     0.159D 00    0.998931     0.000288   1.000347    -0.000203   1.000000   0.000000 -0.420D 01    0.0
5.400000     0.168D 00    0.999016     0.000280   1.000290    -0.000178   1.000000   0.000000 -0.449D 01    0.0
5.700000     0.178D 00    0.999099     0.000277   1.000240    -0.000157   1.000000   0.000000 -0.479D 01    0.0
6.000000     0.187D 00    0.999182     0.000276   1.000196    -0.000139   1.000000   0.000000 -0.508D 01    0.0
6.300000     0.197D 00    0.999265     0.000275   1.000157    -0.000121   1.000000   0.000000 -0.537D 01    0.0
6.600000     0.206D 00    0.999347     0.000274   1.000123    -0.000105   1.000000   0.000000 -0.567D 01    0.0
6.900000     0.215D 00    0.999429     0.000273   1.000094    -0.000090   1.000000   0.000000 -0.596D 01    0.0
7.200000     0.225D 00    0.999511     0.000273   1.000069    -0.000076   1.000000   0.000000 -0.626D 01    0.0
```

```
7.500000    0.234D 00    0.999593    0.000272    1.000048   -0.000062    1.000000    0.000000 -0.655D 01   0.0
7.800000    0.243D 00    0.999675    0.000272    1.000031   -0.000050    1.000000    0.000000 -0.684D 01   0.0
8.100000    0.253D 00    0.999756    0.000271    1.000016   -0.000037    1.000000    0.0      -0.714D 01   0.0
8.400000    0.262D 00    0.999838    0.000271    1.000009   -0.000026    1.000000    0.0      -0.743D 01   0.0
8.700000    0.271D 00    0.999919    0.000271    1.000003   -0.000015    1.000000    0.000000 -0.773D 01   0.0
9.000000    0.281D 00    1.000000    0.000270    1.000000   -0.000004    1.000000    0.0      -0.802D 01   0.0

                                         *****         *****         *****

     XO    = 0.0              R     = 0.0                 PHI  =140.00 DEG.
     XI    = 0.0              DXI   = 0.0                 CWALL= 0.0              NIT  =  2
     HX    = 0.100000D 01     HW    = 0.0

     NONDIMENSIONAL EDGE PROPERTIES

     TE    = 0.300300D 03     UE    = 0.0                 VE    = 0.0
     DUEDX= 0.445732D 01      DVEDX=-0.309671D-01         DUEDW= 0.0                  DVEDW= 0.0

     LOCAL EDGE REYNOLDS NUMBER =0.0

     NONDIMENSIONAL BOUNDARY LAYER PARAMETERS

     CFXINF= 0.0              CFXEDG= 0.0                 CFWINF= 0.0                 CFWEDG= 0.0
     QW    =-0.769736D-02     CHIMAX= 0.0

     DIMENSIONAL BOUNDARY LAYER PARAMETERS

     LONGITUDINAL SKIN FRICTION= 0.0          PSF     DELTA*(X)   = 0.276252D-01     THETA(X)   = 0.120177D-01
     TRANSVERSE SKIN FRICTION  = 0.0          PSF     DELTA*(PHI)= 0.192198D-01     THETA(PHI)= 0.950648D-02
     WALL HEAT TRANSFER RATE   =-0.757259D 00 BTU     DELTA (FT)  = 0.961279D-01

  ETA           Y            F            FN          G            GN           T            TN          V             EPLUS

0.0          0.0          0.0          0.916828    0.0          1.197140    0.999834    0.000086  0.0          0.0
0.300000     0.100D-01    0.252595     0.767450    0.318785     0.930242    0.999860    0.000085 -0.344D-01    0.0
0.600000     0.201D-01    0.460735     0.621592    0.559778     0.683885    0.999885    0.000083 -0.131D 00    0.0
0.900000     0.301D-01    0.626173     0.484070    0.731856     0.473609    0.999909    0.000079 -0.278D 00    0.0
1.200000     0.402D-01    0.752200     0.360150    0.847252     0.307018    0.999932    0.000070 -0.466D 00    0.0
1.500000     0.502D-01    0.843638     0.254454    0.919428     0.184899    0.999951    0.000059 -0.683D 00    0.0
1.800000     0.603D-01    0.906463     0.169797    0.961186     0.102497    0.999967    0.000047 -0.923D 00    0.0
2.100000     0.703D-01    0.947131     0.106518    0.983299     0.051660    0.999979    0.000034 -0.118D 01    0.0
2.400000     0.804D-01    0.971832     0.062580    0.993858     0.023233    0.999988    0.000023 -0.144D 01    0.0
2.700000     0.904D-01    0.985859     0.034333    0.998294     0.009007    0.999993    0.000014 -0.171D 01    0.0
3.000000     0.100D 00    0.993290     0.017555    0.999851     0.002767    0.999997    0.000008 -0.198D 01    0.0
3.300000     0.111D 00    0.996957     0.008363    1.000241     0.000466    0.999998    0.000004 -0.225D 01    0.0
3.600000     0.121D 00    0.998644     0.003720    1.000250    -0.000168    0.999999    0.000002 -0.253D 01    0.0
3.900000     0.131D 00    0.999371     0.001559    1.000180    -0.000232    1.000000    0.000001 -0.280D 01    0.0
4.200000     0.141D 00    0.999669     0.000630    1.000119    -0.000163    1.000000    0.000000 -0.308D 01    0.0
4.500000     0.151D 00    0.999790     0.000261    1.000081    -0.000098    1.000000    0.000000 -0.335D 01    0.0
4.800000     0.161D 00    0.999842     0.000123    1.000058    -0.000059    1.000000    0.000000 -0.363D 01    0.0
5.100000     0.171D 00    0.999869     0.000073    1.000044    -0.000038    1.000000    0.000000 -0.390D 01    0.0
5.400000     0.181D 00    0.999888     0.000054    1.000035    -0.000028    1.000000    0.000000 -0.418D 01    0.0
5.700000     0.191D 00    0.999902     0.000045    1.000027    -0.000022    1.000000    0.000000 -0.445D 01    0.0
6.000000     0.201D 00    0.999915     0.000040    1.000022    -0.000018    1.000000    0.000000 -0.473D 01    0.0
6.300000     0.211D 00    0.999926     0.000036    1.000017    -0.000015    1.000000    0.000000 -0.500D 01    0.0
6.600000     0.221D 00    0.999937     0.000033    1.000013    -0.000012    1.000000    0.000000 -0.528D 01    0.0
6.900000     0.231D 00    0.999946     0.000031    1.000010    -0.000010    1.000000    0.000000 -0.555D 01    0.0
7.200000     0.241D 00    0.999955     0.000029    1.000007    -0.000008    1.000000    0.000000 -0.583D 01    0.0
7.500000     0.251D 00    0.999963     0.000027    1.000005    -0.000006    1.000000    0.000000 -0.610D 01    0.0
7.800000     0.261D 00    0.999971     0.000026    1.000003    -0.000005    1.000000    0.000000 -0.637D 01    0.0
8.100000     0.271D 00    0.999979     0.000025    1.000002    -0.000004    1.000000    0.000000 -0.665D 01    0.0
8.400000     0.281D 00    0.999986     0.000024    1.000001    -0.000002    1.000000    0.000000 -0.692D 01    0.0
8.700000     0.291D 00    0.999993     0.000023    1.000000    -0.000001    1.000000    0.000000 -0.720D 01    0.0
```

```
9.000000   0.301D 00   1.000000   0.000022   1.000000  -0.000000   1.000000  -0.000000 -0.747D 01   0.0
                                            *****         *****          *****

     X0    = 0.0               R     = 0.0                PHI   =160.00 DEG.
     XI    = 0.0               DXI   = 0.0                CWALL= 0.0               NIT   =  3
     HX    = 0.1000000D 01     HW    = 0.0

     NONDIMENSIONAL EDGE PROPERTIES

     TE    = 0.3003000D 03     UE    = 0.0                VE    = 0.0
     DUEDX= 0.349046D 01       DVEDX= 0.1100150D 00       DUEDW= 0.0                    DVEDW= 0.0

     LOCAL EDGE REYNOLDS NUMBER =0.0

     NONDIMENSIONAL BOUNDARY LAYER PARAMETERS

     CFXINF= 0.0               CFXEDG= 0.0                CFWINF= 0.0                   CFWEDG= 0.0
     QW     =-0.7203830-02     CHIMAX= 0.0

     DIMENSIONAL BOUNDARY LAYER PARAMETERS

     LONGITUDINAL SKIN FRICTION= 0.0            PSF     DELTA*(X)    = 0.3033310-01     THETA(X)    = 0.1311830-01
     TRANSVERSE SKIN FRICTION   = 0.0           PSF     DELTA*(PHI)= 0.2035000-01       THETA(PHI)= 0.1012310-01
     WALL HEAT TRANSFER RATE    =-0.7087060 00 BTU     DELTA (FT)  = 0.1041060 00

  ETA          Y           F           FN          G           GN          T          TN          V            EPLUS

0.0         0.0         0.0          0.931703   0.0          1.267378   0.999834   0.000091   0.0            0.0
0.300000    0.114D-01   0.256978     0.781579   0.335052     0.968981   0.999861   0.000090  -0.407D-01      0.0
0.600000    0.227D-01   0.469061     0.633171   0.583416     0.696045   0.999888   0.000088  -0.155D 00      0.0
0.900000    0.341D-01   0.637338     0.490981   0.756035     0.467256   0.999913   0.000082  -0.328D 00      0.0
1.200000    0.454D-01   0.764595     0.361378   0.867742     0.290865   0.999937   0.000072  -0.546D 00      0.0
1.500000    0.568D-01   0.855585     0.250572   0.934475     0.166322   0.999956   0.000059  -0.799D 00      0.0
1.800000    0.681D-01   0.916667     0.162609   0.970893     0.086326   0.999972   0.000045  -0.107D 01      0.0
2.100000    0.795D-01   0.954935     0.098233   0.988796     0.039991   0.999983   0.000031  -0.136D 01      0.0
2.400000    0.908D-01   0.977203     0.055007   0.996559     0.016076   0.999991   0.000020  -0.166D 01      0.0
2.700000    0.102D 00   0.989192     0.028457   0.999411     0.005278   0.999995   0.000011  -0.197D 01      0.0
3.000000    0.114D 00   0.995149     0.013573   1.000213     0.001158   0.999998   0.000006  -0.228D 01      0.0
3.300000    0.125D 00   0.997876     0.005966   1.000314    -0.000064   0.999999   0.000003  -0.258D 01      0.0
3.600000    0.136D 00   0.999028     0.002426   1.000247    -0.000260   1.000000   0.000001  -0.289D 01      0.0
3.900000    0.148D 00   0.999481     0.000927   1.000175    -0.000198   1.000000   0.000001  -0.320D 01      0.0
4.200000    0.159D 00   0.999651     0.000350   1.000129    -0.000120   1.000000   0.000000  -0.351D 01      0.0
4.500000    0.170D 00   0.999717     0.000148   1.000101    -0.000074   1.000000   0.000000  -0.382D 01      0.0
4.800000    0.182D 00   0.999749     0.000084   1.000082    -0.000052   1.000000   0.000000  -0.413D 01      0.0
5.100000    0.193D 00   0.999771     0.000065   1.000069    -0.000041   1.000000   0.000000  -0.444D 01      0.0
5.400000    0.204D 00   0.999789     0.000060   1.000057    -0.000035   1.000000   0.000000  -0.475D 01      0.0
5.700000    0.216D 00   0.999807     0.000059   1.000048    -0.000031   1.000000   0.000000  -0.506D 01      0.0
6.000000    0.227D 00   0.999825     0.000059   1.000039    -0.000027   1.000000   0.000000  -0.537D 01      0.0
6.300000    0.238D 00   0.999842     0.000058   1.000031    -0.000024   1.000000   0.000000  -0.568D 01      0.0
6.600000    0.250D 00   0.999860     0.000058   1.000025    -0.000021   1.000000   0.000000  -0.599D 01      0.0
6.900000    0.261D 00   0.999877     0.000058   1.000019    -0.000018   1.000000   0.000000  -0.630D 01      0.0
7.200000    0.273D 00   0.999895     0.000058   1.000014    -0.000015   1.000000   0.000000  -0.661D 01      0.0
7.500000    0.284D 00   0.999912     0.000058   1.000010    -0.000012   1.000000   0.000000  -0.692D 01      0.0
7.800000    0.295D 00   0.999930     0.000058   1.000006    -0.000010   1.000000   0.000000  -0.723D 01      0.0
8.100000    0.307D 00   0.999947     0.000059   1.000004    -0.000008   1.000000   0.0        -0.754D 01      0.0
8.400000    0.318D 00   0.999965     0.000059   1.000002    -0.000005   1.000000   0.0        -0.785D 01      0.0
8.700000    0.329D 00   0.999982     0.000059   1.000001    -0.000003   1.000000   0.000000  -0.816D 01      0.0
9.000000    0.341D 00   1.000000     0.000059   1.000000    -0.000001   1.000000   0.000000  -0.847D 01      0.0
                                            *****         *****          *****
```

```
XO    = 0.0                R     = 0.0                PHI   =180.00 DEG.
XI    = 0.0                DXI   = 0.0                CWALL= 0.0              NIT  =  3
HX    = 0.1000000 01       HW    = 0.0

NONDIMENSIONAL EDGE PROPERTIES

TE    = 0.300300D 03       UE    = 0.0                VE    = 0.0
DUEDX= 0.308012D 01        DVEDX= 0.261478D-15        DUEDW= 0.0                     DVEDW= 0.0

LOCAL EDGE REYNOLDS NUMBER =0.0

NONDIMENSIONAL BOUNDARY LAYER PARAMETERS

CFXINF= 0.0                CFXEDG= 0.0                     CFWINF= 0.0                    CFWEDG= 0.0
QW     =-0.664662D-02      CHIMAX= 0.0

DIMENSIONAL BOUNDARY LAYER PARAMETERS

LONGITUDINAL SKIN FRICTION= 0.0             PSF     DELTA*(X)   = 0.325431D-01     THETA(X)   = 0.140710D-01
TRANSVERSE SKIN FRICTION  = 0.0             PSF     DELTA*(PHI)= 0.363760D-01      THETA(PHI)= 0.156391D-01
WALL HEAT TRANSFER RATE    =-0.653888D 00 BTU     DELTA (FT) = 0.126104D 00

  ETA          Y            F            FN           G            GN           T            TN           V            FPLUS

0.0          0.0          0.0          0.926670     0.0          0.606461     0.999834     0.000089  0.0           0.0
0.300000     0.121D-01    0.255522     0.776962     0.181923     0.603551     0.999860     0.000089 -0.388D-01     0.0
0.600000     0.242D-01    0.466335     0.629497     0.360132     0.576361     0.999887     0.000086 -0.147D 00     0.0
0.900000     0.363D-01    0.633728     0.488902     0.525181     0.515961     0.999912     0.000081 -0.311D 00     0.0
1.200000     0.483D-01    0.760645     0.361206     0.667679     0.428987     0.999935     0.000071 -0.518D 00     0.0
1.500000     0.604D-01    0.851855     0.252092     0.781656     0.329837     0.999955     0.000059 -0.757D 00     0.0
1.800000     0.725D-01    0.913581     0.165190     0.865848     0.233970     0.999970     0.000045 -0.102D 01     0.0
2.100000     0.846D-01    0.952695     0.101112     0.923177     0.152881     0.999982     0.000032 -0.130D 01     0.0
2.400000     0.967D-01    0.975797     0.057569     0.959113     0.091908     0.999990     0.000021 -0.158D 01     0.0
2.700000     0.109D 00    0.988466     0.030383     0.979826     0.050780     0.999995     0.000012 -0.187D 01     0.0
3.000000     0.121D 00    0.994896     0.014822     0.990794     0.025760     0.999997     0.000007 -0.217D 01     0.0
3.300000     0.133D 00    0.997910     0.006667     0.996122     0.011985     0.999999     0.000003 -0.246D 01     0.0
3.600000     0.145D 00    0.999210     0.002759     0.998496     0.005109     1.000000     0.000002 -0.276D 01     0.0
3.900000     0.157D 00    0.999725     0.001048     0.999464     0.001993     1.000000     0.000001 -0.305D 01     0.0
4.200000     0.169D 00    0.999913     0.000364     0.999825     0.000710     1.000000     0.000000 -0.335D 01     0.0
4.500000     0.181D 00    0.999975     0.000115     0.999948     0.000231     1.000000     0.000000 -0.364D 01     0.0
4.800000     0.193D 00    0.999994     0.000033     0.999986     0.000068     1.000000     0.000000 -0.394D 01     0.0
5.100000     0.205D 00    0.999999     0.000008     0.999997     0.000018     1.000000     0.000000 -0.423D 01     0.0
5.400000     0.218D 00    1.000000     0.000002     0.999999     0.000004     1.000000     0.000000 -0.453D 01     0.0
5.700000     0.230D 00    1.000000     0.000000     1.000000     0.000001     1.000000     0.000000 -0.482D 01     0.0
6.000000     0.242D 00    1.000000    -0.000000     1.000000     0.000000     1.000000     0.000000 -0.512D 01     0.0
6.300000     0.254D 00    1.000000    -0.000000     1.000000     0.000000     1.000000     0.000000 -0.542D 01     0.0
6.600000     0.266D 00    1.000000    -0.000000     1.000000    -0.000000     1.000000     0.000000 -0.571D 01     0.0
6.900000     0.278D 00    1.000000    -0.000000     1.000000    -0.000000     1.000000     0.000000 -0.601D 01     0.0
7.200000     0.290D 00    1.000000    -0.000000     1.000000    -0.000000     1.000000     0.000000 -0.630D 01     0.0
7.500000     0.302D 00    1.000000    -0.000000     1.000000    -0.000000     1.000000     0.000000 -0.660D 01     0.0
7.800000     0.314D 00    1.000000    -0.000000     1.000000    -0.000000     1.000000     0.000000 -0.689D 01     0.0
8.100000     0.326D 00    1.000000    -0.000000     1.000000    -0.000000     1.000000     0.000000 -0.719D 01     0.0
8.400000     0.338D 00    1.000000    -0.000000     1.000000    -0.000000     1.000000     0.000000 -0.749D 01     0.0
8.700000     0.351D 00    1.000000    -0.000000     1.000000    -0.000000     1.000000     0.000000 -0.778D 01     0.0
9.000000     0.363D 00    1.000000    -0.000000     1.000000    -0.000000     1.000000     0.000000 -0.808D 01     0.0

                                                *****               *****               *****

XO    = 0.313260D-02       R     = 0.197231D-01       PHI   =  0.0  DEG.
XI    = 0.200000D-01       DXI   = 0.200000D-01       CWALL= 0.0              NIT  =  3
HX    = 0.100000D 01       HW    = 0.123932D 00
```

```
NONDIMENSIONAL EDGE PROPERTIES

TE    = 0.300300D 03     UE    = 0.823882D-01     VE    = 0.0
DUEDX= 0.411906D 01     DVEDX= 0.0               DUEDW= 0.0                      DVEDW= 0.413296D-02

LOCAL EDGE REYNOLDS NUMBER =0.164776D 00

NONDIMENSIONAL BOUNDARY LAYER PARAMETERS

CFXINF= 0.439024D-01      CFXEDG= 0.185648D 01      CFWINF= 0.0                  CFWEDG= 0.0
QW     =-0.850093D-02     CHIMAX= 0.162571D-02

DIMENSIONAL BOUNDARY LAYER PARAMETERS

LONGITUDINAL SKIN FRICTION= 0.167927D 02 PSF     DELTA*(X)   = 0.280463D-01     THETA(X)   = 0.121156D-01
TRANSVERSE SKIN FRICTION  = 0.0          PSF     DELTA*(PHI)= 0.293362D-01     THETA(PHI)= 0.133146D-01
WALL HEAT TRANSFER RATE   =-0.836313D 00 BTU     DELTA (FT) = 0.104272D 00

  ETA          Y            F            FN           G            GN           T            TN          V            EPLUS

0.0        0.0          0.0          0.928241     0.0          0.753358     0.999834     0.000099  0.0           0.0
0.300000   0.105D-01    0.255990     0.778492     0.214103     0.673383     0.999863     0.000093 -0.394D-01     0.0
0.600000   0.209D-01    0.467237     0.630803     0.403582     0.587899     0.999890     0.000088 -0.149D 00     0.0
0.900000   0.314D-01    0.634961     0.489744     0.566229     0.494544     0.999915     0.000080 -0.316D 00     0.0
1.200000   0.418D-01    0.762041     0.361434     0.699889     0.395859     0.999938     0.000070 -0.527D 00     0.0
1.500000   0.523D-01    0.853227     0.251726     0.803822     0.298308     0.999957     0.000057 -0.771D 00     0.0
1.800000   0.627D-01    0.914771     0.164409     0.879564     0.209848     0.999972     0.000044 -0.104D 01     0.0
2.100000   0.732D-01    0.953617     0.100166     0.930914     0.136934     0.999983     0.000031 -0.132D 01     0.0
2.400000   0.836D-01    0.976437     0.056681     0.963129     0.082507     0.999990     0.000020 -0.161D 01     0.0
2.700000   0.941D-01    0.988865     0.029686     0.981756     0.045752     0.999995     0.000012 -0.191D 01     0.0
3.000000   0.105D 00    0.995120     0.014349     0.991656     0.023293     0.999998     0.000006 -0.221D 01     0.0
3.300000   0.115D 00    0.998022     0.006386     0.996482     0.010866     0.999999     0.000003 -0.251D 01     0.0
3.600000   0.125D 00    0.999260     0.002611     0.998636     0.004638     1.000000     0.000001 -0.281D 01     0.0
3.900000   0.136D 00    0.999745     0.000979     0.999514     0.001808     1.000000     0.000001 -0.311D 01     0.0
4.200000   0.146D 00    0.999919     0.000336     0.999842     0.000643     1.000000     0.000000 -0.341D 01     0.0
4.500000   0.157D 00    0.999977     0.000106     0.999953     0.000208     1.000000     0.000000 -0.371D 01     0.0
4.800000   0.167D 00    0.999994     0.000030     0.999987     0.000061     1.000000     0.000000 -0.401D 01     0.0
5.100000   0.178D 00    0.999998     0.000008     0.999997     0.000016     1.000000     0.000000 -0.431D 01     0.0
5.400000   0.188D 00    1.000000     0.000002     0.999999     0.000004     1.000000     0.000000 -0.461D 01     0.0
5.700000   0.199D 00    1.000000     0.000000     1.000000     0.000001     1.000000     0.000000 -0.492D 01     0.0
6.000000   0.209D 00    1.000000     0.000000     1.000000     0.000000     1.000000     0.000000 -0.522D 01     0.0
6.300000   0.219D 00    1.000000     0.000000     1.000000     0.000000     1.000000     0.000000 -0.552D 01     0.0
6.600000   0.230D 00    1.000000     0.000000     1.000000     0.000000     1.000000     0.000000 -0.582D 01     0.0
6.900000   0.240D 00    1.000000     0.000000     1.000000     0.000000     1.000000     0.000000 -0.612D 01     0.0
7.200000   0.251D 00    1.000000     0.000000     1.000000     0.000000     1.000000     0.000000 -0.642D 01     0.0
7.500000   0.261D 00    1.000000     0.000000     1.000000     0.000000     1.000000     0.000000 -0.672D 01     0.0
7.800000   0.272D 00    1.000000     0.000000     1.000000     0.000000     1.000000     0.000000 -0.702D 01     0.0
8.100000   0.282D 00    1.000000     0.000000     1.000000     0.000000     1.000000     0.000000 -0.733D 01     0.0
8.400000   0.293D 00    1.000000     0.000000     1.000000     0.000000     1.000000     0.0          -0.763D 01     0.0
8.700000   0.303D 00    1.000000     0.000000     1.000000     0.000000     1.000000     0.000000 -0.793D 01     0.0
9.000000   0.314D 00    1.000000     0.000000     1.000000     0.000000     1.000000    -0.000000 -0.823D 01     0.0

                                                        *****         *****          *****

XO   = 0.306541D-02     R    = 0.197403D-01     PHI  = 20.00 DEG.
XI   = 0.200000D-01     DXI  = 0.200000D-01     CWALL= 0.0                 NIT  =  3
HX   = 0.100023D 01     HW   = 0.124017D 00

NONDIMENSIONAL EDGE PROPERTIES

TE    = 0.300300D 03     UE    = 0.831193D-01     VE    = 0.209946D-03
DUEDX= 0.419005D 01     DVEDX=-0.178396D-02     DUEDW= 0.162137D-01     DVEDW= 0.294792D-02
```

```
LOCAL EDGE REYNOLDS NUMBER =0.166239D 00

NONDIMENSIONAL BOUNDARY LAYER PARAMETERS

CFXINF= 0.446643D-01     CFXEDG= 0.186383D 01     CFWINF= 0.167951D-03     CFWEDG= 0.167951D-03
QW    =-0.857481D-02     CHIMAX= 0.162664D-02

DIMENSIONAL BOUNDARY LAYER PARAMETERS

LONGITUDINAL SKIN FRICTION= 0.170341D 02 PSF     DELTA*(X)   = 0.278221D-01     THETA(X)   = 0.120203D-01
TRANSVERSE SKIN FRICTION  = 0.642411D-01 PSF     DELTA*(PHI)= 0.169828D-01     THETA(PHI)= 0.355463D-02
WALL HEAT TRANSFER RATE   =-0.843581D 00 BTU     DELTA (FT) = 0.103529D 00

  ETA           Y           F           FN          G           GN          T           TN          V          FPLUS

0.0        0.0         0.0         0.931917    0.0         0.003504    0.999834    0.000099   0.0          0.0
0.300000   0.104D-01   0.256903    0.780902    0.000913    0.002591    0.999863    0.000094  -0.400D-01    0.0
0.600000   0.208D-01   0.468687    0.632004    0.001563    0.001781    0.999890    0.000085  -0.150D 00    0.0
0.900000   0.312D-01   0.636609    0.489925    0.001994    0.001134    0.999915    0.000080  -0.318D 00    0.0
1.200000   0.416D-01   0.763621    0.360879    0.002257    0.000665    0.999938    0.000070  -0.530D 00    0.0
1.500000   0.520D-01   0.854567    0.250760    0.002404    0.000354    0.999957    0.000057  -0.774D 00    0.0
1.800000   0.624D-01   0.915794    0.163329    0.002479    0.000168    0.999972    0.000043  -0.104D 01    0.0
2.100000   0.728D-01   0.954326    0.099192    0.002512    0.000069    0.999983    0.000030  -0.132D 01    0.0
2.400000   0.832D-01   0.976884    0.055926    0.002524    0.000022    0.999991    0.000020  -0.161D 01    0.0
2.700000   0.937D-01   0.989122    0.029170    0.002527    0.000004    0.999995    0.000011  -0.191D 01    0.0
3.000000   0.104D 00   0.995255    0.014035    0.002527   -0.000001    0.999998    0.000006  -0.221D 01    0.0
3.300000   0.114D 00   0.998086    0.006214    0.002527   -0.000002    0.999999    0.000003  -0.251D 01    0.0
3.600000   0.125D 00   0.999288    0.002527    0.002526   -0.000001    1.000000    0.000001  -0.281D 01    0.0
3.900000   0.135D 00   0.999755    0.000942    0.002526   -0.000001    1.000000    0.000001  -0.311D 01    0.0
4.200000   0.146D 00   0.999922    0.000321    0.002526   -0.000000    1.000000    0.000000  -0.341D 01    0.0
4.500000   0.156D 00   0.999977    0.000100    0.002526   -0.000000    1.000000    0.000000  -0.371D 01    0.0
4.800000   0.166D 00   0.999993    0.000029    0.002526   -0.000000    1.000000    0.000000  -0.401D 01    0.0
5.100000   0.177D 00   0.999998    0.000008    0.002526   -0.000000    1.000000   -0.000000  -0.432D 01    0.0
5.400000   0.187D 00   0.999999    0.000002    0.002526   -0.000000    1.000000   -0.000000  -0.462D 01    0.0
5.700000   0.198D 00   0.999999    0.000001    0.002526   -0.000000    1.000000   -0.000000  -0.492D 01    0.0
6.000000   0.208D 00   0.999999    0.000000    0.002526   -0.000000    1.000000   -0.000000  -0.522D 01    0.0
6.300000   0.219D 00   0.999999    0.000000    0.002526   -0.000000    1.000000   -0.000000  -0.552D 01    0.0
6.600000   0.229D 00   1.000000    0.000000    0.002526   -0.000000    1.000000   -0.000000  -0.582D 01    0.0
6.900000   0.239D 00   1.000000    0.000000    0.002526   -0.000000    1.000000   -0.000000  -0.612D 01    0.0
7.200000   0.250D 00   1.000000    0.000000    0.002526   -0.000000    1.000000   -0.000000  -0.642D 01    0.0
7.500000   0.260D 00   1.000000    0.000000    0.002526   -0.000000    1.000000   -0.000000  -0.672D 01    0.0
7.800000   0.271D 00   1.000000    0.000000    0.002526   -0.000000    1.000000   -0.000000  -0.703D 01    0.0
8.100000   0.281D 00   1.000000    0.000000    0.002526   -0.000000    1.000000   -0.000000  -0.733D 01    0.0
8.400000   0.291D 00   1.000000    0.000000    0.002526   -0.000000    1.000000   -0.000000  -0.763D 01    0.0
8.700000   0.302D 00   1.000000    0.000000    0.002526   -0.000000    1.000000   -0.000000  -0.793D 01    0.0
9.000000   0.312D 00   1.000000    0.000000    0.002526   -0.000000    1.000000   -0.000000  -0.823D 01    0.0

                                                *****       *****       *****

XO   = 0.287100D-02     R    = 0.197886D-01     PHI  = 40.00 DEG.
XI   = 0.200000D-01     DXI  = 0.200000D-01     CWALL= 0.0                 NIT  =  2
HX   = 0.100090D 01     HW   = 0.124345D 00

NONDIMENSIONAL EDGE PROPERTIES

TE    = 0.300300D 03     UE    = 0.841120D-01     VE    = 0.275812D-03
DUEDX= 0.426873D 01     DVEDX=-0.158899D-01     DUEDW= 0.228429D-01     DVEDW=-0.107620D-02

LOCAL EDGE REYNOLDS NUMBER =0.168224D 00

NONDIMENSIONAL BOUNDARY LAYER PARAMETERS
```

```
CFXINF= 0.455498D-01     CFXEDG= 0.185724D 01     CFWINF= 0.209988D-03     CFWEDG= 0.209988D-03
QW     =-0.867212D-02    CHIMAX= 0.163374D-02

DIMENSIONAL BOUNDARY LAYER PARAMETERS

LONGITUDINAL SKIN FRICTION= 0.174228D 02 PSF     DELTA*(X)   = 0.276304D-01     THETA(X)   = 0.119423D-01
TRANSVERSE SKIN FRICTION  = 0.303204D-01 PSF     DELTA*(PHI)= 0.182319D-01     THETA(PHI)= 0.920052D-02
WALL HEAT TRANSFER RATE   =-0.853154D 00 BTU     DELTA (FT) = 0.102905D 00

 ETA          Y           F           FN          G           GN          T           TN          V            FPLUS

0.0         0.0         0.0         0.933618    0.0         0.004304    0.999834    0.000100  0.0           0.0
0.300000    0.103D-01   0.257284    0.781770    0.001129    0.003232    0.999863    0.000094 -0.397D-01     0.0
0.600000    0.207D-01   0.469224    0.632231    0.001948    0.002272    0.999890    0.000083 -0.150D 00     0.0
0.900000    0.310D-01   0.637145    0.489751    0.002506    0.001494    0.999916    0.000080 -0.317D 00     0.0
1.200000    0.414D-01   0.764070    0.360518    0.002860    0.000916    0.999938    0.000070 -0.529D 00     0.0
1.500000    0.517D-01   0.854897    0.250364    0.003069    0.000520    0.999957    0.000057 -0.773D 00     0.0
1.800000    0.621D-01   0.916012    0.162987    0.003183    0.000271    0.999972    0.000043 -0.104D 01     0.0
2.100000    0.724D-01   0.954454    0.098938    0.003239    0.000128    0.999983    0.000030 -0.132D 01     0.0
2.400000    0.828D-01   0.976949    0.055758    0.003265    0.000054    0.999991    0.000020 -0.161D 01     0.0
2.700000    0.931D-01   0.989148    0.029072    0.003275    0.000020    0.999995    0.000011 -0.191D 01     0.0
3.000000    0.103D 00   0.995259    0.013984    0.003276    0.000006    0.999998    0.000006 -0.221D 01     0.0
3.300000    0.114D 00   0.998080    0.006191    0.003279    0.000001    0.999999    0.000003 -0.251D 01     0.0
3.600000    0.124D 00   0.999277    0.002519    0.003279    0.000000    1.000000    0.000001 -0.231D 01     0.0
3.900000    0.134D 00   0.999744    0.000941    0.003279   -0.000000    1.000000    0.000000 -0.311D 01     0.0
4.200000    0.145D 00   0.999912    0.000324    0.003279   -0.000000    1.000000    0.000000 -0.341D 01     0.0
4.500000    0.155D 00   0.999967    0.000104    0.003279   -0.000000    1.000000    0.000000 -0.371D 01     0.0
4.800000    0.166D 00   0.999985    0.000033    0.003279   -0.000000    1.000000   -0.000000 -0.401D 01     0.0
5.100000    0.176D 00   0.999990    0.000012    0.003279   -0.000000    1.000000   -0.000000 -0.431D 01     0.0
5.400000    0.186D 00   0.999993    0.000006    0.003279   -0.000000    1.000000   -0.000000 -0.461D 01     0.0
5.700000    0.197D 00   0.999994    0.000004    0.003279   -0.000000    1.000000   -0.000000 -0.492D 01     0.0
6.000000    0.207D 00   0.999995    0.000003    0.003279   -0.000000    1.000000   -0.000000 -0.522D 01     0.0
6.300000    0.217D 00   0.999996    0.000003    0.003279   -0.000000    1.000000   -0.000000 -0.552D 01     0.0
6.600000    0.228D 00   0.999997    0.000003    0.003279   -0.000000    1.000000   -0.000000 -0.582D 01     0.0
6.900000    0.238D 00   0.999998    0.000002    0.003279   -0.000000    1.000000   -0.000000 -0.612D 01     0.0
7.200000    0.248D 00   0.999998    0.000002    0.003279   -0.000000    1.000000   -0.000000 -0.642D 01     0.0
7.500000    0.259D 00   0.999999    0.000002    0.003279   -0.000000    1.000000   -0.000000 -0.672D 01     0.0
7.800000    0.269D 00   0.999999    0.000001    0.003279   -0.000000    1.000000   -0.000000 -0.702D 01     0.0
8.100000    0.279D 00   0.999999    0.000001    0.003279   -0.000000    1.000000   -0.000000 -0.732D 01     0.0
8.400000    0.290D 00   1.000000    0.000001    0.003279   -0.000000    1.000000   -0.000000 -0.762D 01     0.0
8.700000    0.300D 00   1.000000    0.000000    0.003279   -0.000000    1.000000   -0.000000 -0.793D 01     0.0
9.000000    0.310D 00   1.000000    0.000000    0.003279   -0.000000    1.000000   -0.000000 -0.823D 01     0.0

                                             *****            *****            *****

XO     = 0.257056D-02     R      = 0.198596D-01     PHI   = 60.00 DEG.
XI     = 0.200000D-01     DXI    = 0.200000D-01     CWALL= 0.0                   NIT   =  4
HX     = 0.100195D 01     HW     = 0.124844D 00

NONDIMENSIONAL EDGE PROPERTIES

TE     = 0.300300D 03     UE     = 0.857473D-01     VE     = 0.421988D-04
DUEDX= 0.440017D 01      DVEDX=-0.552826D-01      DUEDW= 0.352476D-01      DVEDW=-0.754545D-02

LOCAL EDGE REYNOLDS NUMBER =0.171495D 00

NONDIMENSIONAL BOUNDARY LAYER PARAMETERS

CFXINF= 0.470382D-01     CFXEDG= 0.187335D 01     CFWINF=-0.150574D-04     CFWEDG=-0.150574D-04
QW     =-0.884210D-02    CHIMAX= 0.164456D-02

DIMENSIONAL BOUNDARY LAYER PARAMETERS
```

```
        LONGITUDINAL SKIN FRICTION= 0.179921D 02 PSF      DELTA*(X)   = 0.273096D-01      THETA(X)   = 0.118118D-01
        TRANSVERSE SKIN FRICTION  =-0.575945D-02 PSF      DELTA*(PHI)= 0.522801D-01      THETA(PHI)= 0.976520D-02
        WALL HEAT TRANSFER RATE   =-0.869377D 00 BTU      DELTA (FT) = 0.101900D 00

    ETA          Y          F          FN          G           GN          T          TN          V         EPLUS

 0.0         0.0        0.0        0.936677   0.0        -0.000300   0.999834   0.000101  0.0          0.0
 0.300000    0.102D-01  0.257993   0.783473  -0.000047   -0.000023   0.999863   0.000095 -0.391D-01    0.0
 0.600000    0.205D-01  0.470269   0.632351  -0.000019    0.000190   0.999891   0.000089 -0.149D 00    0.0
 0.900000    0.307D-01  0.638251   0.489619   0.000059    0.000309   0.999916   0.000081 -0.315D 00    0.0
 1.200000    0.410D-01  0.765059   0.359944   0.000160    0.000340   0.999939   0.000070 -0.528D 00    0.0
 1.500000    0.512D-01  0.855678   0.249607   0.000258    0.000307   0.999958   0.000057 -0.772D 00    0.0
 1.800000    0.615D-01  0.916562   0.162241   0.000341    0.000241   0.999973   0.000043 -0.104D 01    0.0
 2.100000    0.717D-01  0.954795   0.093317   0.000403    0.000168   0.999984   0.000030 -0.132D 01    0.0
 2.400000    0.820D-01  0.977130   0.055309   0.000443    0.000106   0.999991   0.000019 -0.161D 01    0.0
 2.700000    0.922D-01  0.989219   0.028786   0.000468    0.000060   0.999996   0.000011 -0.191D 01    0.0
 3.000000    0.102D 00  0.995265   0.013825   0.000481    0.000031   0.999998   0.000006 -0.220D 01    0.0
 3.300000    0.113D 00  0.998053   0.006118   0.000487    0.000015   0.999999   0.000003 -0.250D 01    0.0
 3.600000    0.123D 00  0.999237   0.002494   0.000490    0.000006   1.000000   0.000001 -0.280D 01    0.0
 3.900000    0.133D 00  0.999701   0.000941   0.000491    0.000002   1.000000   0.000000 -0.311D 01    0.0
 4.200000    0.143D 00  0.999870   0.000335   0.000492    0.000001   1.000000   0.000000 -0.341D 01    0.0
 4.500000    0.154D 00  0.999929   0.000119   0.000492    0.000000   1.000000  -0.000000 -0.371D 01    0.0
 4.800000    0.164D 00  0.999951   0.000048   0.000492    0.000000   1.000000  -0.000000 -0.401D 01    0.0
 5.100000    0.174D 00  0.999962   0.000026   0.000492    0.000000   1.000000  -0.000000 -0.431D 01    0.0
 5.400000    0.184D 00  0.999968   0.000019   0.000492    0.000000   1.000000  -0.000000 -0.461D 01    0.0
 5.700000    0.195D 00  0.999973   0.000016   0.000492    0.000000   1.000000  -0.000000 -0.491D 01    0.0
 6.000000    0.205D 00  0.999978   0.000015   0.000492    0.000000   1.000000  -0.000000 -0.521D 01    0.0
 6.300000    0.215D 00  0.999982   0.000013   0.000492    0.000000   1.000000  -0.000000 -0.551D 01    0.0
 6.600000    0.225D 00  0.999986   0.000012   0.000492    0.000000   1.000000  -0.000000 -0.581D 01    0.0
 6.900000    0.236D 00  0.999989   0.000010   0.000492    0.000000   1.000000  -0.000000 -0.611D 01    0.0
 7.200000    0.246D 00  0.999992   0.000009   0.000492    0.000000   1.000000  -0.000000 -0.641D 01    0.0
 7.500000    0.256D 00  0.999994   0.000007   0.000492    0.000000   1.000000  -0.000000 -0.671D 01    0.0
 7.800000    0.266D 00  0.999996   0.000006   0.000492    0.000000   1.000000  -0.000000 -0.701D 01    0.0
 8.100000    0.277D 00  0.999998   0.000004   0.000492    0.000000   1.000000  -0.000000 -0.731D 01    0.0
 8.400000    0.287D 00  0.999999   0.000003   0.000492    0.000000   1.000000  -0.000000 -0.761D 01    0.0
 8.700000    0.297D 00  1.000000   0.000002   0.000492    0.000000   1.000000  -0.000000 -0.792D 01    0.0
 9.000000    0.307D 00  1.000000   0.000001   0.000492    0.000000   1.000000   0.000000 -0.822D 01    0.0

                                             *****                *****                *****

        XO     = 0.219793D-02     R      = 0.199412D-01     PHI   = 80.00 DEG.
        XI     = 0.200000D-01     DXI    = 0.200000D-01     CWALL= 0.0                 NIT   =  3
        HX     = 0.100326D 01     HW     = 0.125405D 00

        NONDIMENSIONAL EDGE PROPERTIES

        TE     = 0.300300D 03     UE     = 0.878899D-01     VE     =-0.529943D-03
        DUEDX= 0.457560D 01       DVEDX=-0.120707D 00       DUEDW= 0.386691D-01      DVEDW=-0.120687D-01

        LOCAL EDGE REYNOLDS NUMBER =0.175780D 00

        NONDIMENSIONAL BOUNDARY LAYER PARAMETERS

        CFXINF= 0.490110D-01      CFXEDG= 0.188098D 01      CFWINF=-0.550906D-03      CFWEDG=-0.550906D-03
        QW     =-0.901737D-02     CHIMAX= 0.165625D-02

        DIMENSIONAL BOUNDARY LAYER PARAMETERS

        LONGITUDINAL SKIN FRICTION= 0.187467D 02 PSF      DELTA*(X)   = 0.269223D-01      THETA(X)   = 0.116607D-01
        TRANSVERSE SKIN FRICTION  =-0.210722D 00 PSF      DELTA*(PHI)= 0.100988D-01      THETA(PHI)= 0.431052D-02
        WALL HEAT TRANSFER RATE   =-0.887120D 00 BTU      DELTA (FT) = 0.100831D 00
```

APPENDIX IX

Program ICBL3D Daca FORMAT Sheets

IBM

FORTRAN Coding Form

GX28-7327-6 U/M 050**
Printed in U.S.A.

PROGRAM ICBL3D

PROGRAMMER | DATE | PUNCHING INSTRUCTIONS | GRAPHIC | PUNCH | PAGE OF | CARD ELECTRO NUMBER*

STATEMENT NUMBER | CONT | FORTRAN STATEMENT | IDENTIFICATION SEQUENCE

```
DATA DECK

CARD 1
                                      LABEL
 ┌──────────────────────────────────────────────────────────────────────────┐
 └──────────────────────────────────────────────────────────────────────────┘
                                      ( 20A4)
      LABEL    TITLE OF CASE. APPEARS AS THE HEADER TO PROGRAM OUTPUT

CARD 2
                                                      IE
                                                     ┌──┐
                                                     └──┘
                                                   (49X,I3)
      IE       NUMBER OF POINTS NORMAL TO BODY. RECOMMENDED VALUE 101.

CARD 3
                                                     INJCT
                                                     ┌──┐
                                                     └──┘
                                                   (49X,I3)
      INJCT    SUBSCRIPT OF XSTA ARRAY, GIVES LOCATION FOR MASS INJECTION
```

**Number of forms per pad may vary slightly

IBM FORTRAN Coding Form

GX28-7327-6 U/M 050**
Printed in U.S.A.

PROGRAM ICBL3D

PROGRAMMER

DATE

PUNCHING INSTRUCTIONS GRAPHIC

PUNCH

PAGE OF

CARD ELECTRO NUMBER*

STATEMENT NUMBER | FORTRAN STATEMENT | IDENTIFICATION SEQUENCE

```
CARD 4
                                               KADETA
                                                 □
                                              (49X, I3)
      KADETA   INDICATOR FOR THE ADJUSTMENT OF THE TRANSFORMED NORMAL COORDINATE
CARD 5
                                                KEND2
                                                 □
                                              (49X, I3)
      KEND2    NUMBER OF CIRCUMFERENTIAL PLANES
CARD 6
                                               KONSET
                                                 □
                                              (49X, I3)
      KONSET   SUBSCRIPT OF XSTA ARRAY, LOCATES ONSET OF TRANSITION
```

**Number of forms per pad may vary slightly

IBM

FORTRAN Coding Form

GX28-7327-6 U/M 050**
Printed in U.S.A.

PROGRAM ICBL3D

PROGRAMMER

DATE

PUNCHING INSTRUCTIONS

GRAPHIC

PUNCH

PAGE OF

CARD ELECTRO NUMBER*

STATEMENT NUMBER

FORTRAN STATEMENT

IDENTIFICATION SEQUENCE

```
CARD 7
                                                  KPRT
                                                  [  ]
                                               (49X, I3)
        KPRT      PARAMETER WHICH CONTROLS PRINTING OF PROFILES IN Φ DIRECTION
CARD 8
                                                 KTRANS
                                                  [  ]
                                               (49X, I3)
        KTRANS    INDICATOR FOR TRANSITION MODEL
CARD 9
                                                 LAMTRB
                                                  [  ]
                                               (49X, I3)
        LAMTRB    INDICATES WHETHER FLOW IS LAMINAR OR TURBULENT
```

**Number of forms per pad may vary slightly

IBM

FORTRAN Coding Form

GX28-7327-6 U/M 050**
Printed in U.S.A.

PROGRAM ICBL3D

PROGRAMMER

DATE

PUNCHING INSTRUCTIONS GRAPHIC PUNCH

PAGE OF

CARD ELECTRO NUMBER*

STATEMENT NUMBER

FORTRAN STATEMENT

IDENTIFICATION SEQUENCE

```
CARD 10
                                                      LPRT
                                                      [  ]
                                                   (49X, I3)
     LPRT      PRINT CONTROL PARAMETER IN THE STREAMWISE DIRECTION
CARD 11
                                                      NIT1
                                                      [  ]
                                                   (49X, I3)
     NIT1      ITERATION COUNTER USED TO ADJUST STREAMWISE STEP SIZE
CARD 12
                                                      NIT2
                                                      [  ]
                                                   (49X, I3)
     NIT2      ITERATION COUNTER USED TO ADJUST STREAMWISE STEP SIZE
```

IBM FORTRAN Coding Form GX28-7327-6 U/M 050** Printed in U.S.A.

PROGRAM ICBL3D

PROGRAMMER | DATE | PUNCHING INSTRUCTIONS | GRAPHIC | PUNCH | PAGE OF | CARD ELECTRO NUMBER

STATEMENT NUMBER | CONT | FORTRAN STATEMENT | IDENTIFICATION SEQUENCE

```
CARD 13
                                                  NIT3
                                                   □
                                               (49X, I3)
          NIT3    ITERATION COUNTER USED TO ADJUST STREAMWISE STEP SIZE
CARD 14
                                                  NOINJ
                                                   □
                                               (49X, I3)
          NOINJ   SUBSCRIPT OF XSTA ARRAY, LOCATES END OF INJECTION
CARD 15
                                                  SFC
                                                   □
                                               (49X, A3)
          SFC     LITERAL VARIABLE INDICATES IF SURFACE CURVATURE EFFECTS INCLUDED
```

IBM FORTRAN Coding Form

GX28-7327-6 U/M 050**
Printed in U.S.A.

PROGRAM ICBL3D		PUNCHING INSTRUCTIONS	GRAPHIC		PAGE OF
PROGRAMMER	DATE		PUNCH		CARD ELECTRO NUMBER*

STATEMENT NUMBER | FORTRAN STATEMENT | IDENTIFICATION SEQUENCE

```
CARD 16
                                                   NSOLVE
                                                     □
                                                 ( 49 X, I 3 )
        NSOLVE     NUMBER OF VARIABLES IN THE XSTA ARRAY
CARD 17
                                                       ADTEST
                                                   □
                                                      (49X, E14. 6)
        ADTEST     PROVIDES CONVERGENCE CRITERIA  FOR STREAMWISE VELOCITY PROFILE
CARD 18
                                                        AKSTAR
                                                   □
                                                     (4 9X, E14. 6)
        AKSTAR     VAN DRIEST INNER EDDY VISCOSITY LAW CONSTANT (REC. VALUE .435)
```

*Number of forms per pad may vary slightly

IBM FORTRAN Coding Form

GX28-7327-6 U/M 050**
Printed in U.S.A.

PROGRAM ICBL3D

PROGRAMMER | DATE | PUNCHING INSTRUCTIONS | GRAPHIC | PUNCH | PAGE OF | CARD ELECTRO NUMBER*

STATEMENT NUMBER | FORTRAN STATEMENT | IDENTIFICATION SEQUENCE

```
CARD 19
                                                        ALAMDA
                                                  [                ]
                                                   (49X,E14.6)
         ALAMDA    CONSTANT IN OUTER VISCOSITY LAW (REC. VALUE 0.09)
CARD 20
                                                         ASTAR
                                                  [                ]
                                                   (49X,E14.6)
         ASTAR     CONSTANT IN DAMPING TERM VAN DRIEST'S INNER VISCOSITY LAW
                                                                (REC.VALUE 26.0)
CARD 21
                                                         CWALL
                                                  [                ]
                                                   (49X,E14.6)
         CWALL     CONSTANT RATE FOR MASS TRANSFER CASES
```

*Number of forms per pad may vary slightly

IBM

FORTRAN Coding Form

GX28-7327-6 U/M 050**
Printed in U.S.A.

PROGRAM ICBL3D		PUNCHING INSTRUCTIONS	GRAPHIC	PAGE OF
PROGRAMMER	DATE		PUNCH	CARD ELECTRO NUMBER*

FORTRAN STATEMENT

```
CARD 22
                                                        CONV
                                                   [              ]
                                                    (49X,E14.6)
          CONV        CONVERGENCE  CRITERION
CARD 23
                                                        DXMAX
                                                   [              ]
                                                    (49X,E14.6)
          DMAX        MAXIMUM STEP  SIZE
CARD 24
                                                         DX
                                                   [              ]
                                                    (49X,E14.6)
          DX1         INITIAL STREAMWISE STEP  SIZE
```

*Number of forms per pad may vary slightly

IBM

FORTRAN Coding Form

GX28-7327-6 U/M 050**
Printed in U.S.A.

PROGRAM ICBL3D

PROGRAMMER

DATE

PUNCHING INSTRUCTIONS

GRAPHIC

PUNCH

PAGE OF

CARD ELECTRO NUMBER*

STATEMENT NUMBER

CONT

FORTRAN STATEMENT

IDENTIFICATION SEQUENCE

CARD 25

EDYLAW

□

(49X, A3)

EDYLAW INNER EDDY VISCOSITY LAW TO BE USED IN TURBULENT CASES

CARD 26

ETAFAC

□

(49X, E14.6)

ETAFAC CONTROLS NORMAL GRID SPACING

CARD 27

ETAINF

□

(49X, E14.6)

ETAINF MAXIMUM VALUE OF η.

**Number of forms per pad may vary slightly

IBM FORTRAN Coding Form GX28-7327-6 U/M 050**
Printed in U.S.A.

PROGRAM ICBL3D

PROGRAMMER DATE

PUNCHING INSTRUCTIONS GRAPHIC PUNCH PAGE OF CARD ELECTRO NUMBER*

STATEMENT NUMBER FORTRAN STATEMENT IDENTIFICATION SEQUENCE

```
CARD 28
                                                         PR
                                                   [            ]
                                                   (49X, E14.6)
        PR          PRANDTL NUMBER OF FLUID
CARD 29
                                                         RTW
                                                   [            ]
                                                   (49X, E14.6)
        RTW         RATIO WALL TEMPERATURE TO STAGNATION TEMPERATURE
CARD 30
                                                         CP
                                                   [            ]
                                                   (49X, E14.6)
        CP          SPECIFIC HEAT OF FLUID
```

*Number of forms per pad may vary slightly

IBM

FORTRAN Coding Form

GX28-7327-6 U/M 050**
Printed in U.S.A.

PROGRAM ICBL3D

PROGRAMMER

DATE

PUNCHING INSTRUCTIONS

GRAPHIC

PUNCH

PAGE

CARD ELECTRO NUMBER*

```
C  STATEMENT   FORTRAN STATEMENT                                          IDENTIFICATION
   NUMBER                                                                  SEQUENCE

CARD 31
                                                         AMUINF
                                                     [            ]
                                                      (49X,E14.6)
      AMUINF   COEFFICIENT OF VISCOSITY
CARD 32
                                                         PSTAG
                                                     [            ]
                                                      (49X,E14.6)
      PSTAG    FREESTREAM TOTAL PRESSURE
CARD 33
                                                         PINF
                                                     [            ]
                                                      (49X,E14.6)
      PINF     FREESTREAM STATIC PRESSURE
```

**Number of forms per pad may vary slightly

IBM FORTRAN Coding Form

GX28-7327-6 U/M 050**
Printed in U.S.A.

PROGRAM ICBL3D

PROGRAMMER | DATE | PUNCHING INSTRUCTIONS | GRAPHIC | PUNCH | PAGE OF | CARD ELECTRO NUMBER*

FORTRAN STATEMENT | IDENTIFICATION SEQUENCE

```
CARD 34
                                                            XBAR

                                                        (49X, E14.6)
       XBAR      RELATIVE LENGTH OF THE TRANSITION REGIME IN TURBULENT CASES

CARD 35
                                                            UFS

                                                        (49X, E14.6)
       UFS       FREE-STREAM VELOCITY

CARD 36 TO CARD 36+NSOLVE
   XSTA(I) I=1,NSOLVE

    (F12.6)
       XSTA(I)   ARRAY OF SURFACE DISTANCES WHERE SOLUTIONS DESIRED
```

*Number of forms per pad may vary slightly

IBM FORTRAN Coding Form

GX28-7327-6 U/M 050**
Printed in U.S.A.

PROGRAM ICBL3D	PUNCHING INSTRUCTIONS	GRAPHIC	PAGE OF
PROGRAMMER / DATE		PUNCH	CARD ELECTRO NUMBER*

STATEMENT NUMBER | CONT | FORTRAN STATEMENT | IDENTIFICATION SEQUENCE

```
CARD 36+NSOLVE+1 TO LAST CARD
     XTW(I)         TWX(I)         XC1(I)         CIX(I)

                    (4E12.6)
      XTW(I)    ARRAY OF SURFACE DISTANCES CORRESPONDING TO WALL TEMPERATURES
      TWX(I)    WALL TEMPERATURE  AT XTW(I)
      XC1(I)    ARRAY OF SURFACE DISTANCES CORRESPONDING TO INJECTION RATES
      CIX(I)    INJECTION RATE  AT XC1(I)
```

**Number of forms per pad may vary slightly

APPENDIX X

Program ICBL3D Listing

DATE 05/12/77

```
C      PROGRAM MAIN
       IMPLICIT REAL*8(A-H,O-Z)
       REAL*8 NUSE
       COMMON /ASSVAR/ IFL,KBL
       COMMON /CONVRG/ CONV,NIT1,NIT2,NIT3,NIT
       COMMON /DEPVAR/ F(2,101,3),FN(2,101,3),G(2,101,3),GN(2,101,3),T(2,
      1101,3),TN(2,101,3),Y(101),YOL(101)
       COMMON /FINDIF/ A(101),BB(101),B(101),CC(101),DD(101),D(101),E(101
      1),CRI
       COMMON /FRSTRM/ RHOINF,PINF,TFS,UFS,AMUINF,CP,REINF,PR
       COMMON /GEOM/ ALPHA,NOSE,RNOSE,WLST,X,XX,WX
       COMMON /INJECT/ INJCT,NOINJ,MASTRN
       COMMON /INTEGR/ IE,IM,KEND,KEND2,KLX,K,L,NBLNT1,IND,KPRT,LPRT,KPR,
      1LPR
       COMMON /PDECOF/ A0(101),A1(101),A2(101),A3(101),A4(101),A5(101)
       COMMON /STAG/ PSTAG,TSTAG,PNC,QWSTAG
       COMMON /SURFAS/ CWALL,CWIND,PEWIND,VWALL,TWALL,XTW(500),TWX(500),X
      1CI(500),CIX(500),TCONW,KCI,KTW
       COMMON /TMPRTR/ TEMP(101),TP(101),RTW,TB
       COMMON /TRANSN/ KTRANS,KONSET,XIF,CHI2(101),CHIMAX,XBAR
       COMMON /TRBLNT/ ASTAR,AKSTAR,ALAMDA,YSUBL,EVSCTY(101),EDYLAW,EPLUS
      1(101),LAMTRB
       COMMON /WSOLVE/ DW
       COMMON /XICORD/ XI,XXI,DXI,XIOLD
       COMMON /XSOLVE/ XSTA(100),DXMAX,DX,DXOLD,DXI,NSOLVE
       COMMON /ZCOORD/ ETAINF,ETAFAC,ETA(101),DETA(101),ADTEST,KADETA
       CALL ERRSET (208,256,0,0,0,207)
       DEFINE FILE 8(61,6472,L,IFL)
       CALL INPUT
       CALL INIT
       CALL AERO
       CALL OUT1
       CALL CONTRL
       END
```

```
       SUBROUTINE ABCDE (W)
       IMPLICIT REAL*8(A-H,O-Z)
       REAL*8 NUSE
       COMMON /FINDIF/ A(101),BB(101),B(101),CC(101),DD(101),D(101),E(101
      1),CRI
       COMMON /INTEGR/ IE,IM,KEND,KEND2,KLX,KK,LL,NBLNT1,IND,KPRT,LPRT,KP
      1R,LPR
       COMMON /PDECOF/ A0(101),A1(101),A2(101),A3(101),A4(101),A5(101)
       COMMON /WSOLVE/ DW,WCOORD
       COMMON /XICORD/ XI,XXI,DXI,XIOLD
       COMMON /ZCOORD/ ETAINF,ETAFAC,ETA(101),DETA(101),ADTEST,KADETA
       DIMENSION W(2,101,3)
```

DATE 05/12/77

```
C                                                                       ABCD 130
C     SUBROUTINE ABCDE CALCULATES THE COEFFICIENTS OF THE GOVERNING     ABCD 140
C     EQUATIONS IN FINITE-DIFFERENCE FORM                               ABCD 150
C                                                                       ABCD 160
      I=1                                                               ABCD 170
      K=2                                                               ABCD 180
C                                                                       ABCD 190
C                                                                       ABCD 200
      IF (KK.EQ.1.AND.LL.EQ.1) GO TO 60                                 ABCD 210
      IF (KK.EQ.1) GO TO 40                                             ABCD 220
      IF (LL.EQ.1) GO TO 20                                             ABCD 230
C                                                                       ABCD 240
C     COEFFICIENTS FOR THE GENERAL CASE                                 ABCD 250
C                                                                       ABCD 260
      DO 10 J=2,IM                                                      ABCD 270
      DETA1=ETA(J+1)-ETA(J)+ETAFAC**2*(ETA(J)-ETA(J-1))                 ABCD 280
      DETA2=(ETA(J+1)-ETA(J))**2+ETAFAC*(ETA(J)-ETA(J-1))**2            ABCD 290
      A(J)=CRI*(2.0D0*ETAFAC*A0(J)/DETA2-ETAFAC**2*A1(J)/DETA1)         ABCD 300
      CC(J)=CRI*(2.0D0*A0(J)/DETA2+A1(J)/DETA1)                         ABCD 310
      BB(J)=CRI*(-2.0D0*(1.0D0+ETAFAC)*A0(J)/DETA2-(1.0D0-ETAFAC**2)*A1(ABCD 320
     1J)/DETA1+A2(J))                                                   ABCD 330
      B(J)=BB(J)+A4(J)/DXI+A5(J)/2.0D0/DW                               ABCD 340
      DD(J)=-(1.0D0-CRI)/CRI*(A(J)*W(I,J-1,K)+BB(J)*W(I,J,K)+CC(J)*W(I,JABCD 350
     1+1,K))                                                            ABCD 360
      D(J)=DD(J)-A3(J)+A4(J)*W(I,J,K)/DXI+A5(J)*(W(I+1,J,K-1)-W(I,J,K+1)ABCD 370
     1+W(I,J,K))/2.0D0/DW                                               ABCD 380
   10 CONTINUE                                                          ABCD 390
      RETURN                                                            ABCD 400
   20 CONTINUE                                                          ABCD 410
C                                                                       ABCD 420
C     COEFFICIENTS FOR THE STAGNATION LINE                              ABCD 430
C                                                                       ABCD 440
      I=2                                                               ABCD 450
      K=1                                                               ABCD 460
      DO 30 J=2,IM                                                      ABCD 470
      DETA1=ETA(J+1)-ETA(J)+ETAFAC**2*(ETA(J)-ETA(J-1))                 ABCD 480
      DETA2=(ETA(J+1)-ETA(J))**2+ETAFAC*(ETA(J)-ETA(J-1))**2            ABCD 490
      A(J)=CRI*(2.0D0*ETAFAC*A0(J)/DETA2-ETAFAC**2*A1(J)/DETA1)         ABCD 500
      CC(J)=CRI*(2.0D0*A0(J)/DETA2+A1(J)/DETA1)                         ABCD 510
      BB(J)=CRI*(-2.0D0*(1.0D0+ETAFAC)*A0(J)/DETA2-(1.0D0-ETAFAC**2)*A1(ABCD 520
     1J)/DETA1+A2(J))                                                   ABCD 530
      B(J)=BB(J)+A5(J)/DW                                               ABCD 540
      D(J)=-A3(J)+A5(J)*W(I,J,K)/DW-(1.0D0-CRI)/CRI*(A(J)*W(I,J-1,K)+BB(ABCD 550
     1J)*W(I,J,K)+CC(J)*W(I,J+1,K))                                     ABCD 560
   30 CONTINUE                                                          ABCD 570
      RETURN                                                            ABCD 580
   40 CONTINUE                                                          ABCD 590
C                                                                       ABCD 600
C     COEFFICIENTS FOR THE WINDWARD STREAMLINE                          ABCD 610
C                                                                       ABCD 620
      I=1                                                               ABCD 630
      K=2
      DO 50 J=2,IM
      DETA1=ETA(J+1)-ETA(J)+ETAFAC**2*(ETA(J)-ETA(J-1))
      DETA2=(ETA(J+1)-ETA(J))**2+ETAFAC*(ETA(J)-ETA(J-1))**2
      A(J)=CRI*(2.0D0*ETAFAC*A0(J)/DETA2-ETAFAC**2*A1(J)/DETA1)
      CC(J)=CRI*(2.0D0*A0(J)/DETA2+A1(J)/DETA1)
      BB(J)=CRI*(-2.0D0*(1.0D0+ETAFAC)*A0(J)/DETA2-(1.0D0-ETAFAC**2)*A1(
     1J)/DETA1+A2(J))
      B(J)=BB(J)+A4(J)/DXI
      D(J)=-(1.0D0-CRI)/CRI*(A(J)*W(I,J-1,K)+BB(J)*W(I,J,K)+CC(J)*W(I,J+
     11,K))+A4(J)*W(I,J,K)/DXI-A3(J)
50    CONTINUE
      RETURN
60    CONTINUE
C
C     O.D.E. COEFFICIENTS FOR THE STAGNATION POINT
C
      DO 70 J=2,IM
      DETA1=ETA(J+1)-ETA(J)+ETAFAC**2*(ETA(J)-ETA(J-1))
      DETA2=(ETA(J+1)-ETA(J))**2+ETAFAC*(ETA(J)-ETA(J-1))**2
      A(J)=2.0D0*ETAFAC*A0(J)/DETA2-ETAFAC**2*A1(J)/DETA1
      CC(J)=2.0D0*A0(J)/DETA2+A1(J)/DETA1
      B(J)=-2.0D0*(1.0D0+ETAFAC)*A0(J)/DETA2-(1.0D0-ETAFAC**2)*A1(J)/DET
     1A1+A2(J)
      D(J)=-A3(J)
70    CONTINUE
      RETURN
      END


      SUBROUTINE ADDETA (TST,ASYM,ETAOLD)
      IMPLICIT REAL*8(A-H,O-Z)
      REAL*8 NOSE
      COMMON /DEPVAR/ F(2,101,3),FN(2,101,3),G(2,101,3),GN(2,101,3),T(2,
     1101,3),TN(2,101,3),Y(101),YOL(101)
      COMMON /GEOM/ ALPHA,NOSE,RNOSE,WLST,X,XX,WX
      COMMON /INTEGR/ IE,IM,KEND,KEND2,KLX,K,L,NBLNT1,IND,KPRT,LPRT,KPR,
     1LPR
      COMMON /SOLPNT/ VW(101),GW(101),TW(101),GWN(101),FWN(101),FW(101),
     1TWN(101),XIW,XW,RW
      COMMON /XSOLVE/ XSTA(100),DXMAX,DX,DXOLD,DX1,NSOLVE
      COMMON /ZCOORD/ ETAINF,ETAFAC,ETA0(101),DETA0(101),ADTEST,KADETA
      DIMENSION ETA2(101), DETA2(101), CW2(101), VW2(101), FW2(101), GW2
     1(101), TW2(101), F2A(101), F2B(101), F2C(101), G2A(101), G2B(101),
     2 G2C(101), T2A(101), T2B(101), T2C(101)
      DIMENSION F2D(101), T2D(101), G2D(101), ETA(101)
      DATA BLUNT,SHARP/5HBLUNT,5HSHARP/
C
```

DATE 05/12/77

```
C
      IE2=(IE-1)/2
      IF (TST.EQ.0.D0) GO TO 20
      IF (TST.EQ.2.D0) GO TO 10
      ETAIN2=1.1D0*ETAINF
      WRITE (6,280) X,ETAINF,ETAIN2,TST
      GO TO 20
10    ETAIN2=.9D0*ETAINF
      WRITE (6,290) X,ETAINF,ETAIN2,TST
      GO TO 20
      ETAIN2=ETAINF
      WRITE (6,300) X,ETAOLD,ETAIN2,TST
      GO TO 40
20    CONTINUE
      IF (ETAFAC.EQ.1.0D0) DETA1=ETAIN2/DFLOAT(IM)
      IF (ETAFAC.NE.1.0D0) DETA1=ETAIN2*(ETAFAC-1.0D0)/(ETAFAC**IM-1.0D0
     1)
      DETA2(1)=0.0D0
      DETA2(2)=DETA1
      ETA(1)=0.0D0
      ETA2(1)=0.0D0
      ETA2(2)=DETA2(2)
      DO 30 N=2,IM
      DETA2(N+1)=DETA2(N)*ETAFAC
      ETA2(N+1)=ETA2(N)+DETA2(N+1)
      ETA(N)=ETAO(N)
30    CONTINUE
      ETA2(IE)=ETAIN2
      ETA(IE)=ETAINF
      GO TO 60
40    CONTINUE
      IF (ETAFAC.EQ.1.0D0) DETA1=ETAOLD/DFLOAT(IM)
      IF (ETAFAC.NE.1.0D0) DETA1=ETAOLD*(ETAFAC-1.0D0)/(ETAFAC**IM-1.0D0
     1)
      DETA2(1)=0.0D0
      DETA2(2)=DETA1
      ETA2(1)=0.0D0
      ETA(1)=0.0D0
      ETA(2)=DETA2(2)
      DO 50 N=2,IM
      DETA2(N+1)=DETA2(N)*ETAFAC
      ETA(N+1)=ETA(N)+DETA2(N+1)
      ETA2(N)=ETAO(N)
50    CONTINUE
      ETA(IE)=ETAOLD
      ETA2(IE)=ETAINF
60    DO 160 N=1,IE
      IF (ETA2(N).GE.ETAOLD) GO TO 150
      J=0
70    J=J+1
      IF (ETA2(N).GT.ETA(J)) GO TO 70
      IF (J.LT.2) J=2                                                   ADDE 700
      IF (J.GT.(IE-1)) J=IE-1                                           ADDE 710
      IF (TST.GT.2.0D0) GO TO 90                                        ADDE 720
      CALL INTER3 (ETA2(N),ETA(J-1),ETA(J),ETA(J+1),VW(J-1),VW(J),VW(J+1ADDE 730
     1),VW2(N))                                                         ADDE 740
      CALL INTER3 (ETA2(N),ETA(J-1),ETA(J),ETA(J+1),FW(J-1),FW(J),FW(J+1ADDE 750
     1),FW2(N))                                                         ADDE 760
      CALL INTER3 (ETA2(N),ETA(J-1),ETA(J),ETA(J+1),TW(J-1),TW(J),TW(J+1ADDE 770
     1),TW2(N))                                                         ADDE 780
      IF (GW(IE2).EQ.0.0D0) GO TO 80                                    ADDE 790
      CALL INTER3 (ETA2(N),ETA(J-1),ETA(J),ETA(J+1),GW(J-1),GW(J),GW(J+1ADDE 800
     1),GW2(N))                                                         ADDE 810
80    CONTINUE                                                          ADDE 820
90    IF (TST.EQ.3.0D0) GO TO 110                                       ADDE 830
      CALL INTER3 (ETA2(N),ETA(J-1),ETA(J),ETA(J+1),F(1,J-1,3),F(1,J,3),ADDE 840
     1F(1,J+1,3),F2C(N))                                                ADDE 850
      CALL INTER3 (ETA2(N),ETA(J-1),ETA(J),ETA(J+1),T(1,J-1,3),T(1,J,3),ADDE 860
     1T(1,J+1,3),T2C(N))                                                ADDE 870
      IF (G(1,IE2,3).EQ.0.0D0) GO TO 100                                ADDE 880
      CALL INTER3 (ETA2(N),ETA(J-1),ETA(J),ETA(J+1),G(1,J-1,3),G(1,J,3),ADDE 890
     1G(1,J+1,3),G2C(N))                                                ADDE 900
100   CONTINUE                                                          ADDE 910
110   IF (TST.GT.2.D0) GO TO 140                                        ADDE 920
      CALL INTER3 (ETA2(N),ETA(J-1),ETA(J),ETA(J+1),F(2,J-1,2),F(2,J,2),ADDE 930
     1F(2,J+1,2),F2D(N))                                                ADDE 940
      CALL INTER3 (ETA2(N),ETA(J-1),ETA(J),ETA(J+1),T(2,J-1,2),T(2,J,2),ADDE 950
     1T(2,J+1,2),T2D(N))                                                ADDE 960
      IF (G(2,IE2,2).EQ.0.0D0) GO TO 120                                ADDE 970
      CALL INTER3 (ETA2(N),ETA(J-1),ETA(J),ETA(J+1),G(2,J-1,2),G(2,J,2),ADDE 980
     1G(2,J+1,2),G2D(N))                                                ADDE 990
      IF (TST.EQ.5.D0) GO TO 160                                        ADDE1000
120   CONTINUE                                                          ADDE1010
      CALL INTER3 (ETA2(N),ETA(J-1),ETA(J),ETA(J+1),F(2,J-1,1),F(2,J,1),ADDE1020
     1F(2,J+1,1),F2A(N))                                                ADDE1030
      CALL INTER3 (ETA2(N),ETA(J-1),ETA(J),ETA(J+1),T(2,J-1,1),T(2,J,1),ADDE1040
     1T(2,J+1,1),T2A(N))                                                ADDE1050
      IF (G(2,IE2,1).EQ.0.0D0) GO TO 130                                ADDE1060
      CALL INTER3 (ETA2(N),ETA(J-1),ETA(J),ETA(J+1),G(2,J-1,1),G(2,J,1),ADDE1070
     1G(2,J+1,1),G2A(N))                                                ADDE1080
130   CONTINUE                                                          ADDE1090
140   IF (TST.EQ.4.0D0) GO TO 160                                       ADDE1100
      CALL INTER3 (ETA2(N),ETA(J-1),ETA(J),ETA(J+1),F(1,J-1,2),F(1,J,2),ADDE1110
     1F(1,J+1,2),F2B(N))                                                ADDE1120
      CALL INTER3 (ETA2(N),ETA(J-1),ETA(J),ETA(J+1),T(1,J-1,2),T(1,J,2),ADDE1130
     1T(1,J+1,2),T2B(N))                                                ADDE1140
      IF (G(1,IE2,2).EQ.0.0D0) GO TO 160                                ADDE1150
      CALL INTER3 (ETA2(N),ETA(J-1),ETA(J),ETA(J+1),G(1,J-1,2),G(1,J,2),ADDE1160
     1G(1,J+1,2),G2B(N))                                                ADDE1170
      GO TO 160                                                         ADDE1180
150   CONTINUE                                                          ADDE1190
      VW2(N)=VW(IE)                                                     ADDE1200
      FW2(N)=1.0D0
      GW2(N)=GW(IE)
      TW2(N)=1.0D0
      F2A(N)=1.0D0
      F2B(N)=1.0D0
      F2C(N)=1.0D0
      F2D(N)=1.0D0
      G2A(N)=G(2,IE,1)
      G2C(N)=G(1,IE,3)
      G2B(N)=G(1,IE,2)
      G2D(N)=G(2,IE,2)
      T2B(N)=1.0D0
      T2C(N)=1.0D0
      T2A(N)=1.0D0
      T2D(N)=1.0D0
160   CONTINUE
      IF (TST.NE.2.0D0) GO TO 170
      FW2(IE)=1.0D0
      GW2(IE)=GW(IE)
      TW2(IE)=1.0D0
      F2A(IE)=1.0D0
      F2B(IE)=1.0D0
      F2C(IE)=1.0D0
      F2D(IE)=1.0D0
      G2A(IE)=G(2,IE,1)
      G2B(IE)=G(1,IE,2)
      G2C(IE)=G(1,IE,3)
      G2D(IE)=G(2,IE,2)
      T2A(IE)=1.0D0
      T2B(IE)=1.0D0
      T2C(IE)=1.0D0
      T2D(IE)=1.0D0
170   CONTINUE
      ETAINF=ETAIN2
      DO 220 J=1,IE
      IF (TST.GT.2.0D0) GO TO 180
      ETA0(J)=ETA2(J)
      DETA0(J)=DETA2(J)
      VW(J)=VW2(J)
      FW(J)=FW2(J)
      TW(J)=TW2(J)
      IF (GW(IE2).EQ.0.0D0) GO TO 180
      GW(J)=GW2(J)
180   IF (TST.EQ.4.0D0) GO TO 190
      F(1,J,2)=F2B(J)
      T(1,J,2)=T2B(J)
      IF (G(1,IE2,2).EQ.0.0D0) GO TO 190
      G(1,J,2)=G2B(J)
190   CONTINUE
      IF (TST.GT.2.0D0) GO TO 200
      F(2,J,1)=F2A(J)
      T(2,J,1)=T2A(J)                                                   ADDE1720
      IF (G(2,IE2,1).EQ.0.0D0) GO TO 200                                ADDE1730
      G(2,J,1)=G2A(J)                                                   ADDE1740
200   IF (TST.EQ.3.0D0) GO TO 210                                       ADDE1750
      F(1,J,3)=F2C(J)                                                   ADDE1760
      T(1,J,3)=T2C(J)                                                   ADDE1770
      IF (G(1,IE2,3).EQ.0.0D0) GO TO 210                                ADDE1780
      G(1,J,3)=G2C(J)                                                   ADDE1790
210   IF (TST.GT.2.0D0) GO TO 220                                       ADDE1800
      F(2,J,2)=F2D(J)                                                   ADDE1810
      T(2,J,2)=T2D(J)                                                   ADDE1820
      IF (G(2,IE2,2).EQ.0.0D0) GO TO 220                                ADDE1830
      G(2,J,2)=G2D(J)                                                   ADDE1840
220   CONTINUE                                                          ADDE1850
      IF (TST.GT.2.0D0) GO TO 230                                       ADDE1860
      CALL DERIV (FW,ETA2,IE,1,FWN)                                     ADDE1870
      CALL DERIV (TW,ETA2,IE,1,TWN)                                     ADDE1880
      IF (GW(IE2).NE.0.0D0) CALL DERIV (GW,ETA2,IE,1,GWN)               ADDE1890
230   IF (TST.EQ.4.0D0) GO TO 240                                       ADDE1900
      CALL DERIV3 (F,1,2,ETA2,IE,1,FN)                                  ADDE1910
      CALL DERIV3 (T,1,2,ETA2,IE,1,TN)                                  ADDE1920
      IF (G(1,IE2,2).NE.0.0D0) CALL DERIV3 (G,1,2,ETA2,IE,1,GN)         ADDE1930
240   IF (TST.GT.2.0D0) GO TO 250                                       ADDE1940
      CALL DERIV3 (F,2,1,ETA2,IE,1,FN)                                  ADDE1950
      CALL DERIV3 (T,2,1,ETA2,IE,1,TN)                                  ADDE1960
      IF (G(2,IE2,1).NE.0.0D0) CALL DERIV3 (G,2,1,ETA2,IE,1,GN)         ADDE1970
250   IF (TST.EQ.3.0D0) GO TO 260                                       ADDE1980
      CALL DERIV3 (F,1,3,ETA2,IE,1,FN)                                  ADDE1990
      CALL DERIV3 (T,1,3,ETA2,IE,1,TN)                                  ADDE2000
      IF (G(1,IE2,3).NE.0.0D0) CALL DERIV3 (G,1,3,ETA2,IE,1,GN)         ADDE2010
260   IF (TST.GT.2.0D0) GO TO 270                                       ADDE2020
      CALL DERIV3 (F,2,2,ETA2,IE,1,FN)                                  ADDE2030
      CALL DERIV3 (T,2,2,ETA2,IE,1,TN)                                  ADDE2040
      IF (G(2,IE2,2).NE.0.0D0) CALL DERIV3 (G,2,2,ETA2,IE,1,GN)         ADDE2050
270   CONTINUE                                                          ADDE2060
      RETURN                                                            ADDE2070
C                                                                       ADDE2080
C                                                                       ADDE2090
C                                                                       ADDE2100
280   FORMAT (10X,22HETAINF INCREASED AT X=,F10.5,13H  OLD ETAINF=,F10.5ADDE2110
     1,13H  NEW ETAINF=,F10.5,5X,6HTST = ,F4.1/)                        ADDE2120
290   FORMAT (10X,22HETAINF DECREASED AT X=,F10.5,13H  OLD ETAINF=,F10.5ADDE2130
     1,13H  NEW ETAINF=,F10.5,5X,6HTST = ,F4.1/)                        ADDE2140
300   FORMAT (10X,24HETAOLD INCREASED AT X = ,F10.5,2X,13HOLD ETAINF = ,ADDE2150
     1F10.5,2X,13HNEW ETAINF = ,F10.5,2X,6HTST = ,F4.1/)                ADDE2160
      END                                                               ADDE2170
```

```
DATE  05/12/77

      SUBROUTINE AERO                                                   AERO  10
      IMPLICIT REAL*8 (A-H,O-Z)                                         AERO  20
      COMMON /FRSTRM/ RHOINF,PINF,TFS,UFS,AMUINF,CP,REINF,PR            AERO  30
      COMMON /REF/ PREF,TREF,AMUREF                                     AERO  40
      COMMON /STAG/ PSTAG,TSTAG,PNC,QWSTAG                              AERO  50
      DATA ROIN/4HRHOI/,PIN/4HPINF/                                     AERO  60
C                                                                       AERO  70
C     RHOINF IS IN SLUGS                                                AERO  80
C     PINF IS IN PSIA                                                   AERO  90
C     PSTAG IS IN PSIA                                                  AERO 100
C     PREF IS IN PSIA                                                   AERO 110
C     CP IN FT**2/SEC**2/DEG.R                                          AERO 120
C     R IN FT**2/SEC**2/DEG.R                                           AERO 130
C     UFS IN FT/SEC                                                     AERO 140
C     TWALL,TREF,TSTAG IN DEG.R                                         AERO 150
C     AMUREF IS IN (LB-SEC)/FT**2                                       AERO 160
C                                                                       AERO 170
      AMUREF=AMUINF                                                     AERO 180
      RHOINF=2.D0*(PSTAG-PINF)/UFS**2                                   AERO 190
      REINF=RHOINF*UFS/AMUINF                                           AERO 200
      TSTAG=TFS+.5D0/CP*UFS**2                                          AERO 210
      RETURN                                                            AERO 220
      END                                                               AERO 230


      SUBROUTINE BLUNT2 (ISNT)                                          BLUN  10
      IMPLICIT REAL*8 (A-H,O-Z)                                         BLUN  20
      REAL*8 NOSE                                                       BLUN  30
      COMMON /EDGE/ UEDG,TEDG,VEDG,DUEGDX,DUEGDW,D2UEDW,DVEGDX,DVEGDW,REBLUN  40
     1DG,DRDGDX,ZBOD,XEDG                                               BLUN  50
      COMMON /HRWALL/ RO,HXO,HWO,DRODX,DRODW,D2RDXX,D2RDWW,D2RDXW,D3RXXXBLUN  60
     1,D3RWXX,D3RWWX,D3RWWW,DRWWXX,DRWWWX,DHXODX,DHXODW,DHXDWW,DHWODX,DHBLUN  70
     2WODW,XO,DXODX,DXODW,D2XODX,D2XODW,D2XDWX,D3XXWW                   BLUN  80
      COMMON /FRSTRM/ RHOINF,PINF,TFS,UFS,AMUINF,CP,REINF,PR            BLUN  90
      COMMON /GEOM/ ALPHA,NOSE,RNOSE,WLST,X,XX,WX                       BLUN 100
      COMMON /INTEGR/ IE,IM,KEND,KEND2,KLX,K,L,NBLNT1,IND,KPRT,LPRT,KPR,BLUN 110
     1LPR                                                               BLUN 120
      COMMON /OLD/ VOLD(61),CVOLD(61),ROLD(61)                          BLUN 130
      COMMON /STGEDG/ DVEGXW,DVEWXW,DVE2XW,DVEGXX,DVEWXX,DVE2XX,DVGXXW,DBLUN 140
     1VWXXW,DV2XXW,DHWOXW,DHXOXW                                        BLUN 150
      COMMON /OLDEDG/ R3,UE3,DVE3DW                                     BLUN 160
      COMMON /WSOLVE/ DW,W                                              BLUN 170
      COMMON /TVCURV/ TVC                                               BLUN 180
      DIMENSION A(40,2), B(40,2), C(40,2), D(40,2), E(40,2), F(40,2), G(BLUN 190
     140,2), AX(40,2), BX(40,2), CX(40,2), AXX(40,2), DX(40,2), EX(40,2)BLUN 200
     2, USUM(2), RADSUM(2), VSUM(2), VNSUM(2), UNSUM(2), RANSUM(2), HXSUBLUN 210
     3M(2), HWSUM(2), XSFSUM(2), HXNSUM(2), HWNSUM(2), UXSUM(2), VXSUM(2BLUN 220
     4), RXSUM(2), RXXSUM(2), RXNSUM(2), RXNNSM(2), RXXNSM(2), RXXNNS(2)BLUN 230
```

DATE 05/12/77

```
     5, HXXSUM(2), HWXSUM(2), HXNNSM(2), RNNNSM(2), RNNSUM(2), XOSUM(2),BLUN 240
     6 XONSUM(2), XONNSM(2), XOXSUM(2), XUXNSM(2), GX(40,2), RXNNNS(2), BLUN 250
     7XXS(2), UNNSUM(2), VXNSUM(2), HWXNSM(2), HXXNSM(2)                BLUN 260
      DIMENSION ARADS(40), AUS(40), AVS(40), AVXS(40), AHXS(40), AHWS(40BLUN 270
     1), AXSRFS(40), AUX(40), AVX(40), ARX(40), ARXX(40), AHXXS(40), AHWBLUN 280
     2XS(40), AXO(40), AXOX(40)                                          BLUN 290
      DATA YES/3HYES/                                                   BLUN 300
C                                                                       BLUN 310
C                                                                       BLUN 320
      PI=DARCOS(-1.D0)                                                  BLUN 330
      IF (K.NE.1) GO TO 50                                              BLUN 340
      IF (L.EQ.1) IRD=0                                                 BLUN 350
      M=2                                                               BLUN 360
C     *******************************************                       BLUN 370
C     XS = BODY FIXED SURFACE DISTANCE                                  BLUN 380
C     *******************************************                       BLUN 390
C                                                                       BLUN 400
C     AXIAL DISTANCE AND FOURIER COEFFICIENTS ARE READ FROM UNIT 10     BLUN 410
C                                                                       BLUN 420
10    CONTINUE                                                          BLUN 430
      IRD=IRD+1                                                         BLUN 440
      READ (10) XS,ARADS,AUS,AVS,AHXS,AHWS,AXSRFS,AUX,AVX,ARX,ARXX,AHXXSBLUN 450
     1,AHWXS,AXO,AXOX                                                   BLUN 460
      IF (X.GT.XS.OR.IRD.LE.1) GO TO 20                                 BLUN 470
      BACKSPACE 10                                                      BLUN 480
      BACKSPACE 10                                                      BLUN 490
      IRD=IRD-2                                                         BLUN 500
      GO TO 10                                                          BLUN 510
20    CONTINUE                                                          BLUN 520
      IRD=IRD+1                                                         BLUN 530
      READ (10) XS,ARADS,AUS,AVS,AHXS,AHWS,AXSRFS,AUX,AVX,ARX,ARXX,AHXXSBLUN 540
     1,AHWXS,AXO,AXOX                                                   BLUN 550
      IF (X.GT.XS) GO TO 20                                             BLUN 560
      DO 30 J=1,KLX                                                     BLUN 570
      A(J,2)=ARADS(J)                                                   BLUN 580
      B(J,2)=AUS(J)                                                     BLUN 590
      C(J,2)=AVS(J)                                                     BLUN 600
      D(J,2)=AHXS(J)                                                    BLUN 610
      E(J,2)=AHWS(J)                                                    BLUN 620
      F(J,2)=AXSRFS(J)                                                  BLUN 630
      G(J,2)=AXO(J)                                                     BLUN 640
      BX(J,2)=AUX(J)                                                    BLUN 650
      CX(J,2)=AVX(J)                                                    BLUN 660
      AX(J,2)=ARX(J)                                                    BLUN 670
      AXX(J,2)=ARXX(J)                                                  BLUN 680
      DX(J,2)=AHXXS(J)                                                  BLUN 690
      EX(J,2)=AHWXS(J)                                                  BLUN 700
      GX(J,2)=AXOX(J)                                                   BLUN 710
30    CONTINUE                                                          BLUN 720
      XXS(2)=XS                                                         BLUN 730
      BACKSPACE 10                                                      BLUN 740
      BACKSPACE 10
      IRD=IRD-1
      READ (10) XS,ARADS,AUS,AVS,AHXS,AHWS,AXSRFS,AUX,AVX,ARX,ARXX,AHXXS
     1,AHWXS,AXO,AXOX
      DO 40 J=1,KLX
      A(J,1)=ARADS(J)
      B(J,1)=AUS(J)
      C(J,1)=AVS(J)
      D(J,1)=AHXS(J)
      E(J,1)=AHWS(J)
      F(J,1)=AXSRFS(J)
      G(J,1)=AXO(J)
      BX(J,1)=AUX(J)
      CX(J,1)=AVX(J)
      AX(J,1)=ARX(J)
      AXX(J,1)=ARXX(J)
      DX(J,1)=AHXXS(J)
      EX(J,1)=AHWXS(J)
      GX(J,1)=AXOX(J)
   40 CONTINUE
      XXS(1)=XS
   50 CONTINUE
      MM=M-1
      PHI=2.D0*PI*W
      KKL=KLX-1
C
C     FOURIER SERIES ARE USED TO COMPUTE THE EDGE PROPERTIES
C
      DO 80 I=1,M
      USUM(I)=0.0D0
      RADSUM(I)=0.0D0
      VSUM(I)=0.0D0
      HXSUM(I)=0.D0
      HWSUM(I)=0.D0
      XOSUM(I)=0.D0
      XSFSUM(I)=0.D0
      UXSUM(I)=0.D0
      VXSUM(I)=0.D0
      VXNSUM(I)=0.D0
      RXSUM(I)=0.D0
      XOXSUM(I)=0.D0
      RXXSUM(I)=0.D0
      UNSUM(I)=0.0D0
      UNNSUM(I)=0.D0
      RANSUM(I)=0.0D0
      RNNSUM(I)=0.D0
      VNSUM(I)=0.0D0
      HXNSUM(I)=0.D0
      HWNSUM(I)=0.D0
      RXNSUM(I)=0.D0
      RXNNSM(I)=0.D0
      RXNNNS(I)=0.DO
      RXXNSM(I)=0.DO
      RXXNNS(I)=0.DO
      HXXSUM(I)=0.DO
      HWXSUM(I)=0.DO
      HWXNSM(I)=0.DO
      HXXNSM(I)=0.DO
      HXNNSM(I)=0.DO
      RNNNSM(I)=0.DO
      XONSUM(I)=0.DO
      XONNSM(I)=0.DO
      XOXNSM(I)=0.DO
      DO 60 J=1,KLX
      Z=DFLOAT(J)-1.0DO
      SZP=DSIN(Z*PHI)
      CZP=DCOS(Z*PHI)
      RADSUM(I)=RADSUM(I)+A(J,I)*CZP
      XOSUM(I)=XOSUM(I)+G(J,I)*CZP
      USUM(I)=USUM(I)+B(J,I)*CZP
      HXSUM(I)=HXSUM(I)+D(J,I)*CZP
      HWSUM(I)=HWSUM(I)+E(J,I)*CZP
      XSFSUM(I)=XSFSUM(I)+F(J,I)*CZP
      RANSUM(I)=RANSUM(I)-A(J,I)*Z*SZP
      UNSUM(I)=UNSUM(I)-B(J,I)*Z*SZP
      HXNSUM(I)=HXNSUM(I)-D(J,I)*Z*SZP
      HWNSUM(I)=HWNSUM(I)-E(J,I)*Z*SZP
      XONSUM(I)=XONSUM(I)-G(J,I)*Z*SZP
      RNNSUM(I)=RNNSUM(I)-A(J,I)*Z**2*CZP
      XONNSM(I)=XONNSM(I)-G(J,I)*Z**2*CZP
      RXSUM(I)=RXSUM(I)+AX(J,I)*CZP
      UXSUM(I)=UXSUM(I)+BX(J,I)*CZP
      XOXSUM(I)=XOXSUM(I)+GX(J,I)*CZP
      RXNSUM(I)=RXNSUM(I)-AX(J,I)*Z*SZP
      XOXNSM(I)=XOXNSM(I)-GX(J,I)*Z*SZP
      RXNNSM(I)=RXNNSM(I)-AX(J,I)*Z**2*CZP
      RXNNNS(I)=RXNNNS(I)+AX(J,I)*Z**3*SZP
      RXXNSM(I)=RXXNSM(I)-AXX(J,I)*Z*SZP
      RXXNNS(I)=RXXNNS(I)-AXX(J,I)*Z**2*CZP
      HXXSUM(I)=HXXSUM(I)+DX(J,I)*CZP
      HWXSUM(I)=HWXSUM(I)+EX(J,I)*CZP
      IF (L.EQ.1) HWXNSM(I)=HWXNSM(I)-EX(J,I)*Z*SZP
      IF (L.EQ.1) HXXNSM(I)=HXXNSM(I)-DX(J,I)*Z*SZP
      HXNNSM(I)=HXNNSM(I)-D(J,I)*Z**2*CZP
      RXXSUM(I)=RXXSUM(I)+AXX(J,I)*CZP
      RNNNSM(I)=RNNNSM(I)+A(J,I)*Z**3*SZP
      UNNSUM(I)=UNNSUM(I)-B(J,I)*Z**2*CZP
60    CONTINUE
      DO 70 J=1,KKL
      H=DFLOAT(J)
      CHP=DCOS(H*PHI)
      SHP=DSIN(H*PHI)
      VSUM(I)=VSUM(I)+C(J,I)*SHP
      VNSUM(I)=VNSUM(I)+C(J,I)*H*CHP
      VXSUM(I)=VXSUM(I)+CX(J,I)*SHP
      VXNSUM(I)=VXNSUM(I)+CX(J,I)*H*CHP
70    CONTINUE
80    CONTINUE
      IF (K.GT.1) GO TO 90
      VSUM(MM)=0.0D0
      VSUM(M)=0.0D0
      RANSUM(MM)=0.D0
      RANSUM(M)=0.D0
      VXSUM(MM)=0.D0
      VXSUM(M)=0.D0
90    CONTINUE
      IF (MM.EQ.M) GO TO 100
C
C     INTERPOLATION FOR EDGE PROPERTY VALUES AT X
C
      FAC=(X-XXS(MM))/(XXS(M)-XXS(MM))
      GO TO 110
100   FAC=0.0D0
110   CONTINUE
      REDG=RADSUM(MM)+FAC*(RADSUM(M)-RADSUM(MM))
      RO=REDG
      XO=XOSUM(MM)+FAC*(XOSUM(M)-XOSUM(MM))
      XEDG=XO
      TEDG=TFS/UFS**2*CP
      UEDG=USUM(MM)+FAC*(USUM(M)-USUM(MM))
      VEDG=VSUM(MM)+FAC*(VSUM(M)-VSUM(MM))
      HXO=HXSUM(MM)+FAC*(HXSUM(M)-HXSUM(MM))
      HWO=HWSUM(MM)+FAC*(HWSUM(M)-HWSUM(MM))
      XBAR=XSFSUM(MM)+FAC*(XSFSUM(M)-XSFSUM(MM))
      DUEGDW=(UNSUM(MM)+FAC*(UNSUM(M)-UNSUM(MM)))*2.D0*PI
      D2UEDW=(UNNSUM(MM)+FAC*(UNNSUM(M)-UNNSUM(MM)))*4.D0*PI**2
      DVEGDW=(VNSUM(MM)+FAC*(VNSUM(M)-VNSUM(MM)))*2.D0*PI
      DRODW=(RANSUM(MM)+FAC*(RANSUM(M)-RANSUM(MM)))*2.D0*PI
      DXODW=(XONSUM(MM)+FAC*(XONSUM(M)-XONSUM(MM)))*2.D0*PI
      DHXODW=(HXNSUM(MM)+FAC*(HXNSUM(M)-HXNSUM(MM)))*2.D0*PI
      DHWODW=(HWNSUM(MM)+FAC*(HWNSUM(M)-HWNSUM(MM)))*2.D0*PI
      D2RDWW=(RNNSUM(MM)+FAC*(RNNSUM(M)-RNNSUM(MM)))*4.D0*PI**2
      D2XODW=(XONNSM(MM)+FAC*(XONSUM(M)-XONSUM(MM)))*4.D0*PI**2
      D3RWWW=(RNNNSM(MM)+FAC*(RNNNSM(M)-RNNNSM(MM)))*8.D0*PI**3
      DHXDWW=(HXNNSM(MM)+FAC*(HXNNSM(M)-HXNNSM(MM)))*4.D0*PI**2
      IF (L.EQ.1) DHWOXW=HWXNSM(MM)
      IF (L.EQ.1) DHXOXW=HXXNSM(MM)
      DUEGDX=UXSUM(MM)+FAC*(UXSUM(M)-UXSUM(MM))
      DVEGDX=VXSUM(MM)+FAC*(VXSUM(M)-VXSUM(MM))
      DVEGXW=(VXNSUM(MM)+FAC*(VXNSUM(M)-VXNSUM(MM)))*2.D0*PI
      DRODX=RXSUM(MM)+FAC*(RXSUM(M)-RXSUM(MM))
      DXODX=XOXSUM(MM)+FAC*(XOXSUM(M)-XOXSUM(MM))
      DHXODX=HXXSUM(MM)+FAC*(HXXSUM(M)-HXXSUM(MM))
      DHWODX=HWXSUM(MM)+FAC*(HWXSUM(M)-HWXSUM(MM))                      BLUN2280
      D2RDXW=(RXNSUM(MM)+FAC*(RXNSUM(M)-RXNSUM(MM)))*2.D0*PI            BLUN2290
      D2XDWX=(XOXNSM(MM)+FAC*(XOXNSM(M)-XOXNSM(MM)))*2.D0*PI            BLUN2300
      D3RWWX=(RXNNSM(MM)+FAC*(RXNNSM(M)-RXNNSM(MM)))*4.D0*PI**2         BLUN2310
      D2RDXX=RXXSUM(MM)+FAC*(RXXSUM(M)-RXXSUM(MM))                      BLUN2320
      D3RWXX=(RXXNSM(MM)+FAC*(RXXNSM(M)-RXXNSM(MM)))*2.D0*PI            BLUN2330
      DRWWWX=(RXNNNS(MM)+FAC*(RXNNNS(M)-RXNNNS(MM)))*8.D0*PI**3         BLUN2340
      DRWWXX=(RXXNNS(MM)+FAC*(RXXNNS(M)-RXXNNS(MM)))*4.D0*PI**2         BLUN2350
      IF (ISNT.EQ.2) GO TO 130                                          BLUN2360
C                                                                       BLUN2370
C     CALCULATE THE X DERIVATIVES                                       BLUN2380
C                                                                       BLUN2390
      D3RXXX=(RXXSUM(M)-RXXSUM(MM))/(XXS(M)-XXS(MM))                    BLUN2400
      D2XODX=(XOXSUM(M)-XOXSUM(MM))/(XXS(M)-XXS(MM))                    BLUN2410
      D3XXWW=(XONNSM(M)-XONNSM(MM))/(XXS(M)-XXS(MM))                    BLUN2420
      IF (L.NE.1) GO TO 120                                             BLUN2430
      DVEGXX=(VXSUM(M)-VXSUM(MM))/(XXS(M)-XXS(MM))                      BLUN2440
      DVGXXW=(VXNSUM(M)-VXNSUM(MM))/(XXS(M)-XXS(MM))                    BLUN2450
  120 CONTINUE                                                          BLUN2460
      CALL HOFN                                                         BLUN2470
  130 CONTINUE                                                          BLUN2480
      IWT=0                                                             BLUN2490
      IF (IWT.EQ.0) GO TO 140                                           BLUN2500
      WRITE (6,150) UEDG,DUEGDX,DUEGDW,VEDG,DVEGDX,DVEGDW               BLUN2510
      WRITE (6,160) REDG,DRODX,D2RDXX,DRODW,D2RDWW                      BLUN2520
      WRITE (6,170) D2RDXW,D3RWWX,D3RWXX,DRWWXX,DRWWWX                  BLUN2530
      WRITE (6,180) HXO,HWO,DHXODX,DHWODX,DHXODW,DHWODW                 BLUN2540
      WRITE (6,190) XO,DXODX,DXODW                                      BLUN2550
  140 CONTINUE                                                          BLUN2560
      RETURN                                                            BLUN2570
C                                                                       BLUN2580
  150 FORMAT (1H0,7HUEDG = ,E13.6,10H DUEGDX = ,E13.6,10H DUEGDW = ,E13.BLUN2590
     16,7H VEG = ,E13.6,10H DVEGDX = ,E13.6,10H DVEGDW = ,E13.6)        BLUN2600
  160 FORMAT (1H0,7HREDG = ,E13.6,9H DRODX = ,E13.6,10H D2RDXX = ,E13.6,BLUN2610
     110H  DRODW = ,E13.6,10H D2RDWW = ,E13.6)                          BLUN2620
  170 FORMAT (1H0,9HD2RDXW = ,E13.6,10H D3RWWX = ,E13.6,10H D3RWXX = ,E1BLUN2630
     13.6,10H DRWWXX = ,E13.6,10H DRWWWX = ,E13.6)                      BLUN2640
  180 FORMAT (1H0,6HHXO = ,E13.6,7H HWO = ,E13.6,10H DHXODX = ,E13.6,10HBLUN2650
     1 DHWODX = ,E13.6,10H DHXODW = ,E13.6,10H DHWODW = ,E13.6)         BLUN2660
  190 FORMAT (1H0,5HXO = ,E13.6,9H DXODX = ,E13.6,9H DXODW = ,E13.6)    BLUN2670
      END                                                               BLUN2680



      SUBROUTINE CHANGX                                                 CHAN  10
      IMPLICIT REAL*8(A-H,O-Z)                                          CHAN  20
      REAL*8 NOSE                                                       CHAN  30
      COMMON /CONVRG/ CONV,NIT1,NIT2,NIT3,NIT                           CHAN  40
      COMMON /DEPVAR/ F(2,101,3),FN(2,101,3),G(2,101,3),GN(2,101,3),T(2,CHAN  50
     1101,3),TN(2,101,3),Y(101),YOL(101)
      COMMON /GEOM/ ALPHA,NOSE,RNOSE,WLST,X,XX,WX
      COMMON /INJECT/ INJCT,NOINJ,MASTRN
      COMMON /INTEGR/ IE,IM,KEND,KEND2,KLX,K,L,NBLNT1,IND,KPRT,LPRT,KPR,
     1LPR
      COMMON /SOLPNT/ VW(101),GW(101),TW(101),GWN(101),FWN(101),FW(101),
     1TWN(101),XIW,XW,RW
      COMMON /TRANSN/ KTRANS,KONSET,XIF,CHI2(101),CHIMAX,XBAR
      COMMON /TRBLNT/ ASTAR,AKSTAR,ALAMDA,YSUBL,EVSCTY(101),EDYLAW,EPLUS
     1(101),LAMTRB
      COMMON /XSOLVE/ XSTA(100),DXMAX,DX,DXOLD,DX1,NSOLVE
      COMMON /SPECP/ LPRTP
      DATA BLUNT,SHARP/5HBLUNT,5HSHARP/
C
C
      LPRTP=0
      IF (NIT.GE.0) GO TO 30
C
C     CUT BACK X AND DX AND SET DXOLD=DX
C
      DX=DX/2.0D0
      X=X-DX
      DXOLD=DX
C
C     RESET COUNTERS FOR TRANSITION
C
      IF (X.GE.XSTA(KONSET)) GO TO 90
      IF (KONSET.EQ.NSOLVE) GO TO 10
      LAMTRB=1
      XIF=0.0D0
   10 CONTINUE
C
C     RESET COUNTER FOR INJECTION
C
      IF (INJCT.EQ.NSOLVE) GO TO 20
      IF (X.GT.XSTA(INJCT)) GO TO 20
      MASTRN=0
   20 IF (X.GE.XSTA(NOINJ)) MASTRN=0
      RETURN
   30 CONTINUE
C
C     ADJUST DX USING NIT
C
      IF (NIT.GT.NIT1) GO TO 40
      DX=2.0D0*DX
      IF (DX.GT.DXMAX) DX=DXMAX
      GO TO 50
   40 IF (NIT.LT.NIT2) GO TO 50
      DX=0.5D0*DX
   50 CONTINUE
      IF (X.EQ.XSTA(INJCT)) DX=DX/10.0D0
      IF (X.EQ.XSTA(KONSET)) DX=DX/10.0D0                               CHAN 570
      DXOLD=DX                                                          CHAN 580
C                                                                       CHAN 590
C     SET DX TO GIVE A SOLUTION AT XSTA(IXSOLV(II))                     CHAN 600
C                                                                       CHAN 610
      DO 60 I=1,NSOLVE                                                  CHAN 620
      J=I                                                               CHAN 630
      IF (XSTA(I).GT.X.AND.XSTA(I).LE.(X+1.25D0*DX)) GO TO 70           CHAN 640
60    CONTINUE                                                          CHAN 650
      X=X+DX                                                            CHAN 660
      GO TO 80                                                          CHAN 670
70    DX=XSTA(J)-X                                                      CHAN 680
      X=X+DX                                                            CHAN 690
      LPRTP=1                                                           CHAN 700
80    CONTINUE                                                          CHAN 710
C                                                                       CHAN 720
C     BEGIN THE TRANSITION REGIME IF X=XSTA(KONSET)                     CHAN 730
C     CALCULATE THE TRANSITION INTERMITTANCY FACTOR FOR                 CHAN 740
C     X.GE.XSTA(KONSET)                                                 CHAN 750
C                                                                       CHAN 760
      IF (KONSET.EQ.NSOLVE) GO TO 100                                   CHAN 770
      IF (X.LT.XSTA(KONSET)) GO TO 100                                  CHAN 780
      IF (X.GT.XSTA(KONSET)) GO TO 90                                   CHAN 790
      LAMTRB=2                                                          CHAN 800
      WRITE (6,130) X,LAMTRB                                            CHAN 810
90    CONTINUE                                                          CHAN 820
      IF (KTRANS.EQ.0) XIF=1.0D0                                        CHAN 830
      IF (KTRANS.EQ.1) XIF=1.0D0-DEXP(-0.412D0*2.917D0**2*((X-XSTA(KONSECHAN 840
     1T))/(XSTA(KONSET)*(XBAR-1.0D0)))**2)                              CHAN 850
      IF (NIT.LT.0) GO TO 10                                            CHAN 860
C                                                                       CHAN 870
C     BEGIN MASS TRANSFER IF X=XSTA(INJCT)                              CHAN 880
C                                                                       CHAN 890
100   IF (INJCT.EQ.NSOLVE) GO TO 110                                    CHAN 900
      IF (X.LE.XSTA(INJCT)) GO TO 110                                   CHAN 910
      IF (MASTRN.EQ.1) GO TO 110                                        CHAN 920
      MASTRN=1                                                          CHAN 930
      WRITE (6,140) X,MASTRN                                            CHAN 940
C                                                                       CHAN 950
C     END MASS TRANSFER IF X=XSTA(NOINJ)                                CHAN 960
C                                                                       CHAN 970
110   IF (NOINJ.EQ.NSOLVE) GO TO 120                                    CHAN 980
      IF (X.NE.XSTA(NOINJ)) GO TO 120                                   CHAN 990
      MASTRN=0                                                          CHAN1000
      WRITE (6,150) X,MASTRN                                            CHAN1010
120   RETURN                                                            CHAN1020
C                                                                       CHAN1030
C                                                                       CHAN1040
C                                                                       CHAN1050
130   FORMAT (1H0,10X,27HBEGIN TRANSITION REGIME, X=,E12.6,9H  LAMTRB=,ICHAN1060
     12/)                                                               CHAN1070
```

DATE 05/12/77

```
140   FORMAT (1H0,10X,23HBEGIN MASS TRANSFER, X=,E12.6,9H  MASTRN=,I2/)
150   FORMAT (1H0,10X,21HEND MASS TRANSFER, X=,E12.6,9H  MASTRN=,I2/)
      END


      SUBROUTINE CONTRL
      IMPLICIT REAL*8(A-H,O-Z)
      REAL*8 NOSE
      COMMON /ASSVAR/ IFL,KBL
      COMMON /CONICL/ PE(61),TE(61),UE(61),VE(61),DUEDW(61),DVEDW(61),DP
     1EDW2(61)
      COMMON /CONVRG/ CONV,NIT1,NIT2,NIT3,NIT
      COMMON /DEPVAR/ F(2,101,3),FN(2,101,3),G(2,101,3),GN(2,101,3),T(2,
     1101,3),TN(2,101,3),Y(101),YOL(101)
      COMMON /EDGE/ UEDG,TEDG,VEDG,DUEGDX,DUEGDW,D2UEDW,DVEGDX,DVEGDW,RE
     1DG,ZBOO,XEDG
      COMMON /EDGW/ UEW,VEW,TEW,DUEWDX,DUEWDW,D2UEWW,DVEWDX,DVEWDW,REW
      COMMON /EDG2/ TE2,UE2,VE2,DUE2DX,DVE2DX,DUE2DW,D2UE2W,DVE2DW,RE2,X
     12
      COMMON /FINDIF/ A(101),BB(101),B(101),CC(101),DD(101),D(101),E(101
     1),CRI
      COMMON /FRSTRM/ RHOINF,PINF,TFS,UFS,AMUINF,CP,REINF,PR
      COMMON /GASPRP/ PRANDT(101)
      COMMON /GEOM/ ALPHA,NOSE,RNOSE,WLST,X,XX,WX
      COMMON /IECOEF/ B1,B2,B3,B4,B5,B6,AL1,AL2,AL3,XLAMB
      COMMON /INJECT/ INJCT,NOINJ,MASTRN
      COMMON /INTEGR/ IE,IM,KEND,KEND2,KLX,K,L,NBLNT1,IND,KPRT,LPRT,KPR,
     1LPR
      COMMON /OLD/ VOLD(61),CVOLD(61),ROLD(61)
      COMMON /OLDEDG/ R3,UE3,DVE3DW
      COMMON /OUTPUT/ CFWEDG,CFWINF,CFXEDG,CFXINF,DEL,QW,QWINF,QWOQWO,S,
     1TAUETA,TAUX,DELSTX,DELPHI,THETAX,THEPHI,REX2
      COMMON /PDECOF/ A0(101),A1(101),A2(101),A3(101),A4(101),A5(101)
      COMMON /PDEREF/ UREF
      COMMON /SOLPNT/ VW(101),GW(101),TW(101),GWN(101),FWN(101),FW(101),
     1TWN(101),XIW,XW,RW
      COMMON /STAG/ PSTAG,TSTAG,PNC,QWSTAG
      COMMON /SURFAS/ CWALL,CWIND,PEWIND,VWALL,TWALL,XTW(500),TWX(500),X
     1CI(500),CIX(500),TCONW,KCI,KTW
      COMMON /TMPRTR/ TEMP(101),TP(101),RTW,TB
      COMMON /TRBLNT/ ASTAR,AKSTAR,ALAMDA,YSUBL,EVSCTY(101),EDYLAW,EPLUS
     1(101),LAMTRB
      COMMON /WSOLVE/ DW,W
      COMMON /XICORD/ XI,XXI,DXI,XIOLD
      COMMON /XSOLVE/ XSTA(100),DXMAX,DX,DXOLD,DX1,NSOLVE
      COMMON /ZCOORD/ ETAINF,ETAFAC,ETA(101),DETA(101),ADTEST,KADETA
      COMMON /SPECP/ LPRTP
      COMMON /RAD/ RXS(2,101,3)
```

```
DATE  05/12/77

      DIMENSION FOLD(101), GOLD(101), TOLD(101), TS(101,2)
      DIMENSION GNN(2,101,2)
      DATA BLUNT,SHARP/5HBLUNT,5HSHARP/
      NITTOT=0
      ASYM=1.0D0
      ETA2SV=0.0D0
C
C     BEGIN THE LOOP FOR STEPPING DOWNSTREAM
C
10    CONTINUE
      L=L+1
      IFL=1
      FIND(8'IFL)
      W=0.0D0
      IF (KEND.EQ.1.OR.ALPHA.LT.1.D-08) GO TO 20
      DW=WLST/DFLOAT(KEND2-1)
20    CONTINUE
      KPR=KPRT
      IF (L.EQ.1) KLAST=KEND
C
C     BEGIN THE DO-LOOP FOR STEPPING AROUND THE BODY
C
      DO 400 K=1,KEND
      CALL TIMEON
      LC=3
      IF (L.EQ.1) LC=1
      IF (K.EQ.1.AND.L.GT.1) LC=2
      IF (K.EQ.KEND2.AND.L.GT.1) LC=2
      LADETA=0
      WX=W/WLST*180.0D0
      IF (KLAST.GT.0) GO TO 30
      WRITE (6,440)
      RETURN
30    IF (K.GT.KLAST) GO TO 370
40    CONTINUE
C
C     OBTAIN EDGE,AND WALL INVISCID VALUES AND SET
C     MIDPOINT VALUES TO VALUES OBTAINED FOR THIS PLANE AT
C     THE LAST STREAMWISE STATION
C
      CALL EGPROP (LC)
      CALL WALL
      DO 50 J=1,IE
      FW(J)=F(1,J,2)
      GW(J)=G(1,J,2)
      GVEL=GW(J)
      IF (K.EQ.1) GVEL=0.0D0
      TW(J)=T(1,J,2)
      FWN(J)=FN(1,J,2)
      GWN(J)=GN(1,J,2)
      TWN(J)=TN(1,J,2)
```

```
50    CONTINUE
      IF (L.EQ.1) CALL VCALC (LC)
C
C     SAVE THE PROFILES FROM THE LAST ITERATION
C
60    DO 70 J=1,IE
      FOLD(J)=F(2,J,2)
      GOLD(J)=G(2,J,2)
      TOLD(J)=T(2,J,2)
70    CONTINUE
C
C     SOLVE THE STREAMWISE MOMENTUM CONSERVATION EQUATION
C
      IF (LAMTRB.EQ.2) CALL EDYVIS
      CALL XMOM (LC)
      CALL ABCDE (F)
      CALL SOLVE (F,FN,0.0D0,0.0D0,1.0D0)
      IF (L.GT.1) GO TO 90
      DO 80 J=1,IE
      F(1,J,3)=F(2,J,2)*2.0D0-F(2,J,1)
      FN(1,J,3)=FN(2,J,2)*2.0D0-FN(2,J,1)
      F(1,J,2)=F(2,J,2)
      FN(1,J,2)=FN(2,J,2)
80    CONTINUE
90    CONTINUE
      IF (K.GT.1) GO TO 110
      DO 100 J=1,IE
      F(2,J,1)=F(2,J,2)
      F(1,J,3)=F(1,J,2)
      FN(2,J,1)=FN(2,J,2)
      FN(1,J,3)=FN(1,J,2)
100   CONTINUE
110   CONTINUE
      DO 120 J=1,IE
      IF (K.EQ.KLAST) F(1,J,3)=F(2,J,2)-F(2,J,1)+F(1,J,2)
      IF (K.EQ.KLAST) FN(1,J,3)=FN(2,J,2)-FN(2,J,1)+FN(1,J,2)
      FW(J)=F(2,J,2)*CRI+F(1,J,2)*(1.0D0-CRI)
      FWN(J)=FN(2,J,2)*CRI+FN(1,J,2)*(1.0D0-CRI)
      IF (L.EQ.1) FW(J)=F(2,J,1)*(1.0D0-CRI)+F(2,J,2)*CRI
      IF (L.EQ.1) FWN(J)=FN(2,J,1)*(1.0D0-CRI)+FN(2,J,2)*CRI
120   CONTINUE
      IF (ALPHA.LT.1.D-08) GO TO 180
C
C     SOLVE THE CROSSFLOW MOMENTUM CONSERVATION EQUATION
C
      CALL VCALC (LC)
      IF (LAMTRB.EQ.2) CALL EDYVIS
      CALL PHIMOM (LC)
      CALL ABCDE (G)
      CALL SOLVE (G,GN,0.0D0,0.0D0,AL3)
      IF (L.GT.1) GO TO 140
      DO 130 J=1,IE
      G(1,J,3)=G(2,J,2)*2.0D0-G(2,J,1)
      GN(1,J,3)=GN(2,J,2)*2.0D0-GN(2,J,1)
      G(1,J,2)=G(2,J,2)
      GN(1,J,2)=GN(2,J,2)
130   CONTINUE
140   CONTINUE
      IF (K.GT.1) GO TO 160
      DO 150 J=1,IE
      G(2,J,1)=G(2,J,2)
      G(1,J,3)=G(1,J,2)
      GN(2,J,1)=GN(2,J,2)
      GN(1,J,3)=GN(1,J,2)
150   CONTINUE
160   CONTINUE
      DO 170 J=1,IE
      IF (K.EQ.KLAST) G(1,J,3)=G(2,J,2)-G(2,J,1)+G(1,J,2)
      IF (K.EQ.KLAST) GN(1,J,3)=GN(2,J,2)-GN(2,J,1)+GN(1,J,2)
      GW(J)=G(2,J,2)*CRI+(1.0D0-CRI)*G(1,J,2)
      GWN(J)=GN(2,J,2)*CRI+(1.0D0-CRI)*GN(1,J,2)
      IF (L.EQ.1) GW(J)=G(2,J,1)*(1.0D0-CRI)+G(2,J,2)*CRI
      IF (L.EQ.1) GWN(J)=GN(2,J,1)*(1.0D0-CRI)+GN(2,J,2)*CRI
170   CONTINUE
180   CONTINUE
      CALL VCALC (LC)
      NIT=NIT+1
C
C     THE SOLUTION IS CHECKED FOR CONVERGENCE
C
      IF (NIT.LE.NIT3) GO TO 200
      WRITE (6,420) K,L,NIT
      IF (K.GT.1) GO TO 190
      NITTOT=NIT+NITTOT
      IF (NITTOT.GT.(3*NIT3)) WRITE (6,430) K,L,NITTOT
      IF (NITTOT.GT.(3*NIT3)) STOP
      NIT=-1
      CALL CHANGX
      NIT=0
      GO TO 40
190   CONTINUE
      KLAST=K-1
      GO TO 370
200   CONTINUE
C
C     CONVERGENCE TEST ON ALL POINTS OF THE F AND G ARRAYS
C
      DIF=0.0D0
      DO 210 J=2,IE
      DIFF=DABS(F(2,J,2)-FOLD(J))/DABS(FOLD(J))
      IF (DIFF.GT.DIF) DIF=DIFF
      IF (GOLD(J).EQ.0.0D0) GO TO 210
      DIFF=DABS(G(2,J,2)-GOLD(J))/DABS(GOLD(J))                         CONT1970
      IF (DIFF.GT.DIF) DIF=DIFF                                         CONT1980
210   CONTINUE                                                          CONT1990
      IF (DIF.GT.CONV) GO TO 60                                         CONT2000
      IF (NIT.EQ.1) GO TO 60                                            CONT2010
C                                                                       CONT2020
C     SOLVE THE ENERGY CONSERVATION EQUATION                            CONT2030
C                                                                       CONT2040
      IF (LAMTRB.EQ.2) CALL EDYVIS                                      CONT2050
      TB=TWALL                                                          CONT2060
      CALL ENERGY (LC)                                                  CONT2070
      CALL ABCDE (T)                                                    CONT2080
      CALL SOLVE (T,TN,0.0D0,TB,1.0D0)                                  CONT2090
      IF (L.GT.1) GO TO 230                                             CONT2100
      DO 220 J=1,IE                                                     CONT2110
      T(1,J,3)=T(2,J,2)*2.0D0-T(2,J,1)                                  CONT2120
      TN(1,J,3)=TN(2,J,2)*2.0D0-TN(2,J,1)                               CONT2130
      T(1,J,2)=T(2,J,2)                                                 CONT2140
      TN(1,J,2)=TN(2,J,2)                                               CONT2150
220   CONTINUE                                                          CONT2160
230   CONTINUE                                                          CONT2170
      IF (K.GT.1) GO TO 250                                             CONT2180
      DO 240 J=1,IE                                                     CONT2190
      T(2,J,1)=T(2,J,2)                                                 CONT2200
      T(1,J,3)=T(1,J,2)                                                 CONT2210
      TN(2,J,1)=TN(2,J,2)                                               CONT2220
      TN(1,J,3)=TN(1,J,2)                                               CONT2230
240   CONTINUE                                                          CONT2240
250   CONTINUE                                                          CONT2250
      DO 260 J=1,IE                                                     CONT2260
      IF (K.EQ.KLAST) T(1,J,3)=T(2,J,2)-T(2,J,1)+T(1,J,2)               CONT2270
      IF (K.EQ.KLAST) TN(1,J,3)=TN(2,J,2)-TN(2,J,1)+TN(1,J,2)           CONT2280
      TW(J)=T(2,J,2)*CRI+T(1,J,2)*(1.0D0-CRI)                           CONT2290
      TWN(J)=TN(2,J,2)*CRI+TN(1,J,2)*(1.0D0-CRI)                        CONT2300
      IF (L.EQ.1) TW(J)=T(2,J,1)*(1.0D0-CRI)+T(2,J,2)*CRI               CONT2310
      IF (L.EQ.1) TWN(J)=TN(2,J,1)*(1.0D0-CRI)+TN(2,J,2)*CRI            CONT2320
260   CONTINUE                                                          CONT2330
      DO 270 J=1,IE                                                     CONT2340
      GVEL=GW(J)                                                        CONT2350
      IF (K.EQ.1) GVEL=0.0D0                                            CONT2360
270   CONTINUE                                                          CONT2370
C                                                                       CONT2380
C     TEST THE ASYMPTOTIC NATURE OF THE SOLUTION AND ADJUST ETAINF      CONT2390
C     IF NECESSARY                                                      CONT2400
      IF (L.EQ.1) GO TO 300                                             CONT2410
      IF (KADETA.EQ.0) GO TO 300                                        CONT2420
      ASYM=DABS(F(2,IE,2)-F(2,IE-4,2))                                  CONT2430
      IF (ASYM.LT.ADTEST) GO TO 280                                     CONT2440
      TST=1.0D0                                                         CONT2450
      ETAOLD=ETAINF                                                     CONT2460
      GO TO 290                                                         CONT2470
```

```
280    IF (KEND2.GT.1) GO TO 300                                         CONT2480
       IF (ASYM.GT.ADTEST/10.000) GO TO 300                              CONT2490
       TST=2.000                                                         CONT2500
       ETAOLD=ETAINF                                                     CONT2510
290    CONTINUE                                                          CONT2520
       IF (LADETA.NE.0) GO TO 300                                        CONT2530
       LADETA=1                                                          CONT2540
       CALL ADDETA (TST,ASYM,ETAOLD)                                     CONT2550
       GO TO 60                                                          CONT2560
300    CONTINUE                                                          CONT2570
       IF (K.EQ.1.OR.K.EQ.(KEND-1)/2) NITHAF=NIT                         CONT2580
       WRITE (8'IFL) (T(2,J,2),F(2,J,2),G(2,J,2),TN(2,J,2),FN(2,J,2),GN(2CONT2590
      1,J,2),RXS(2,J,2),J=1,IE),ETAINF                                   CONT2600
C                                                                        CONT2610
C      CALCULATE AND WRITE OUT THE RESULTS OF THE CURRENT SOLUTION       CONT2620
C                                                                        CONT2630
       CALL PROPTY                                                       CONT2640
       CALL OUT2                                                         CONT2650
       W=W+DW                                                            CONT2660
C                                                                        CONT2670
C      THE PROGRAM IS SET TO FIND THE SOLUTION AT THE NEXT               CONT2680
C      COORDINATE POINT                                                  CONT2690
C                                                                        CONT2700
       DO 320 J=1,IE                                                     CONT2710
       IF (K.EQ.1.AND.KEND.GT.1) G(2,J,2)=0.000                          CONT2720
       IF (K.EQ.1.AND.KEND.GT.1) GN(2,J,2)=0.000                         CONT2730
       F(2,J,1)=F(2,J,2)                                                 CONT2740
       FN(2,J,1)=FN(2,J,2)                                               CONT2750
       F(1,J,2)=F(1,J,3)                                                 CONT2760
       FN(1,J,2)=FN(1,J,3)                                               CONT2770
       T(2,J,1)=T(2,J,2)                                                 CONT2780
       TN(2,J,1)=TN(2,J,2)                                               CONT2790
       T(1,J,2)=T(1,J,3)                                                 CONT2800
       TN(1,J,2)=TN(1,J,3)                                               CONT2810
       G(2,J,1)=G(2,J,2)                                                 CONT2820
       GN(2,J,1)=GN(2,J,2)                                               CONT2830
       G(1,J,2)=G(1,J,3)                                                 CONT2840
       GN(1,J,2)=GN(1,J,3)                                               CONT2850
       RXS(2,J,1)=RXS(2,J,2)                                             CONT2860
       RXS(1,J,2)=RXS(1,J,3)                                             CONT2870
       IF (L.NE.1) GO TO 310                                             CONT2880
       F(1,J,2)=F(2,J,2)                                                 CONT2890
       FN(1,J,2)=FN(2,J,2)                                               CONT2900
       T(1,J,2)=T(2,J,2)                                                 CONT2910
       TN(1,J,2)=TN(2,J,2)                                               CONT2920
       G(1,J,2)=G(2,J,2)                                                 CONT2930
       GN(1,J,2)=GN(2,J,2)                                               CONT2940
       F(1,J,3)=F(2,J,2)                                                 CONT2950
       FN(1,J,3)=FN(2,J,2)                                               CONT2960
       T(1,J,3)=T(2,J,2)                                                 CONT2970
       TN(1,J,3)=TN(2,J,2)                                               CONT2980
      G(1,J,3)=G(2,J,2)                                                 CONT2990
      GN(1,J,3)=GN(2,J,2)                                               CONT3000
      RXS(1,J,2)=RXS(2,J,2)                                             CONT3010
  310 CONTINUE                                                          CONT3020
  320 CONTINUE                                                          CONT3030
      IF (L.EQ.1) GO TO 370                                             CONT3040
      IF (K.GE.KLAST-1) GO TO 370                                       CONT3050
      READ (8'IFL) (T(1,J,2),F(1,J,2),G(1,J,2),TN(1,J,2),FN(1,J,2),GN(1,CONT3060
     1J,2),RXS(1,J,2),J=1,IE),ETAOLD                                    CONT3070
      TST=3.D0                                                          CONT3080
      IF (ETAOLD.LT.ETAINF) CALL ADDETA (TST,ASYM,ETAOLD)               CONT3090
      READ (8'IFL) (T(1,J,3),F(1,J,3),G(1,J,3),TN(1,J,3),FN(1,J,3),GN(1,CONT3100
     1J,3),RXS(1,J,3),J=1,IE),ETAOLD                                    CONT3110
      TST=4.D0                                                          CONT3120
      IF (ETAOLD.LT.ETAINF) CALL ADDETA (TST,ASYM,ETAOLD)               CONT3130
      IF (L.GT.2) GO TO 340                                             CONT3140
      DO 330 KK=2,3                                                     CONT3150
      DO 330 LL=1,IE                                                    CONT3160
      G(1,LL,KK)=0.D0                                                   CONT3170
      GN(1,LL,KK)=0.D0                                                  CONT3180
  330 CONTINUE                                                          CONT3190
  340 CONTINUE                                                          CONT3200
      IF (K.NE.(KEND2-2)) GO TO 360                                     CONT3210
      DO 350 KK=1,IE                                                    CONT3220
      G(1,KK,3)=0.D0                                                    CONT3230
      GN(1,KK,3)=0.D0                                                   CONT3240
  350 CONTINUE                                                          CONT3250
  360 CONTINUE                                                          CONT3260
      IFL=IFL-2                                                         CONT3270
      FIND(8'IFL)                                                       CONT3280
  370 CONTINUE                                                          CONT3290
      IF (K.NE.KEND) GO TO 390                                          CONT3300
      READ (8'1) (T(1,J,2),F(1,J,2),G(1,J,2),TN(1,J,2),FN(1,J,2),GN(1,J,CONT3310
     12),RXS(1,J,2),J=1,IE),ETAOLD                                      CONT3320
      TST=3.D0                                                          CONT3330
      IF (ETAOLD.LT.ETAINF) CALL ADDETA (TST,ASYM,ETAOLD)               CONT3340
      DO 380 J=1,IE                                                     CONT3350
      T(2,J,1)=T(1,J,2)                                                 CONT3360
      TN(2,J,1)=TN(1,J,2)                                               CONT3370
      T(1,J,3)=T(1,J,2)                                                 CONT3380
      TN(1,J,3)=TN(1,J,2)                                               CONT3390
      F(2,J,1)=F(1,J,2)                                                 CONT3400
      FN(2,J,1)=FN(1,J,2)                                               CONT3410
      F(1,J,3)=F(1,J,2)                                                 CONT3420
      FN(1,J,3)=FN(1,J,2)                                               CONT3430
      G(2,J,1)=G(1,J,2)                                                 CONT3440
      GN(2,J,1)=GN(1,J,2)                                               CONT3450
      G(1,J,3)=G(1,J,2)                                                 CONT3460
      GN(1,J,3)=GN(1,J,2)                                               CONT3470
      RXS(2,J,1)=RXS(1,J,2)                                             CONT3480
      RXS(1,J,3)=RXS(1,J,2)                                             CONT3490
```

```
380     CONTINUE                                                          CONT3500
390     CONTINUE                                                          CONT3510
        NIT=0                                                             CONT3520
        NITTOT=0                                                          CONT3530
        VOLD(K)=UE2                                                       CONT3540
        CVOLD(K)=VE2                                                      CONT3550
        ROLD(K)=RE2                                                       CONT3560
400     CONTINUE                                                          CONT3570
        NIT=NITHAF                                                        CONT3580
        DX=DXOLD                                                          CONT3590
        CALL CHANGX                                                       CONT3600
        NIT=0                                                             CONT3610
        IF (X.GT.XSTA(NSOLVE)) GO TO 410                                  CONT3620
        IF (L.NE.1.AND.L.EQ.LPR.OR.LPRT.EQ.1) LPR=LPR+LPRT                CONT3630
        IF (LPRTP.EQ.1.AND.L.GE.LSAVE) LPR=L+1                            CONT3640
        GO TO 10                                                          CONT3650
410     CONTINUE                                                          CONT3660
        WRITE (6,450)                                                     CONT3670
        RETURN                                                            CONT3680
C                                                                         CONT3690
C                                                                         CONT3700
C                                                                         CONT3710
420     FORMAT (10X,40HFAILED TO GET A CONVERGED SOLUTION AT K=,I3,5X,2HL=CONT3720
       1,I3,5X,4HNIT=,I3)                                                 CONT3730
430     FORMAT (10X,57HEXECUTION TERMINATING******NITTOT.GT.3*NIT3******  CONT3740
       1   K= ,I2,3HL= ,I3,8HNITTOT= ,I3/)                                CONT3750
440     FORMAT (10X,40HEXECUTION TERMINATING******KLAST=0******/)         CONT3760
450     FORMAT (10X,7HTHE END)                                            CONT3770
        END                                                               CONT3780



        SUBROUTINE DERIV (F,X,IMAX,IMIN,FP)                               DERI  10
        IMPLICIT REAL*8 (A-H,O-Z)                                         DERI  20
        DIMENSION F(101), X(101), FP(101)                                 DERI  30
C                                                                         DERI  40
C       SUBROUTINE DERIV3 CALCULATES THE FIRST DERIVATIVES OF F WITH      DERI  50
C       RESPECT TO X AND RETURNS THE ARRAY FP.                            DERI  60
C                                                                         DERI  70
        DO 10 J=IMIN,IMAX                                                 DERI  80
        K=J                                                               DERI  90
        IF (K.LT.(IMIN+1)) K=IMIN+1                                       DERI 100
        IF (K.GT.(IMAX-1)) K=IMAX-1                                       DERI 110
        CALL FD3 (X(J),X(K-1),X(K),X(K+1),F(K-1),F(K),F(K+1),FP(J))       DERI 120
10      CONTINUE                                                          DERI 130
        RETURN                                                            DERI 140
        END                                                               DERI 150
```

```
      SUBROUTINE DERIV3 (FX,II,KK,X,IMAX,IMIN,FPX)                      DER3  10
      IMPLICIT REAL*8 (A-H,O-Z)                                         DER3  20
      REAL*8 NOSE                                                       DER3  30
      COMMON /INTEGR/ IE,IM,KEND,KEND2,KLX,KDUM,L,NBLNT1,IND,KPRT,LPRT,KDER3  40
     1PR,LPR                                                            DER3  50
      DIMENSION X(101), FX(2,101,3), FP(101), F(101), FPX(2,101,3)      DER3  60
C                                                                       DER3  70
C     SUBROUTINE DERIV3 CALCULATES THE FIRST DERIVATIVES OF F WITH      DER3  80
C     RESPECT TO X AND RETURNS THE ARRAY FP.                            DER3  90
C                                                                       DER3 100
      DO 10 J=1,IE                                                      DER3 110
      F(J)=FX(II,J,KK)                                                  DER3 120
   10 CONTINUE                                                          DER3 130
      DO 20 J=IMIN,IMAX                                                 DER3 140
      K=J                                                               DER3 150
      IF (K.LT.(IMIN+1)) K=IMIN+1                                       DER3 160
      IF (K.GT.(IMAX-1)) K=IMAX-1                                       DER3 170
      CALL FD3 (X(J),X(K-1),X(K),X(K+1),F(K-1),F(K),F(K+1),FP(J))       DER3 180
   20 CONTINUE                                                          DER3 190
      DO 30 J=1,IE                                                      DER3 200
      FPX(II,J,KK)=FP(J)                                                DER3 210
   30 CONTINUE                                                          DER3 220
      RETURN                                                            DER3 230
      END                                                               DER3 240


      SUBROUTINE EDGCOF (LC)                                            EDGC  10
      IMPLICIT REAL*8(A-H,O-Z)                                          EDGC  20
      REAL*8 NOSE                                                       EDGC  30
      COMMON /EDGW/ UEW,VEW,TEW,DUEWDX,DUEWDW,D2UEWW,DVEWDX,DVEWDW,REW  EDGC  40
      COMMON /FRSTRM/ RHOINF,PINF,TFS,UFS,AMUINF,CP,REINF,PR            EDGC  50
      COMMON /GEOM/ ALPHA,NOSE,RNOSE,WLST,X,XX,WX                       EDGC  60
      COMMON /IECOEF/ B1,B2,B3,B4,B5,B6,AL1,AL2,AL3,XLAMB               EDGC  70
      COMMON /INTEGR/ IE,IM,KEND,KEND2,KLX,K,L,NBLNT1,IND,KPRT,LPRT,KPR,EDGC  80
     1LPR                                                               EDGC  90
      COMMON /PDEREF/ UREF                                              EDGC 100
      COMMON /SOLPNT/ VW(101),GW(101),TW(101),GWN(101),FWN(101),FW(101),EDGC 110
     1TWN(101),XIW,XW,RW                                                EDGC 120
      COMMON /STAG/ PSTAG,TSTAG,PNC,QWSTAG                              EDGC 130
      COMMON /STGEDG/ DVEGXW,DVEWXW,DVE2XW,DVEGXX,DVEWXX,DVE2XX,DVGXXW,DEDGC 140
     1VWXXW,DV2XXW,DHWOXW,DHXOXW                                        EDGC 150
      COMMON /NCOORD/ XN(101),ETAX,ETAW                                 EDGC 160
      COMMON /XICORD/ XI,XXI,DXI,XIOLD                                  EDGC 170
      COMMON /HETA/ HX(101),HW(101),HXX(101),HXW(101),HXWW(101),HXN(101)EDGC 180
     1,HXNN(101),HWX(101),HWW(101),HWN(101),HWNN(101)                   EDGC 190
```

```
      COMMON /PHI/ DAL3DX,DAL3DW,ALB6                                   EDGC 200
      COMMON /WSOLVE/ DW,W                                              EDGC 210
      DIMENSION AL3OLD(61)                                              EDGC 220
      DATA SHARP,BLUNT/5HSHARP,5HBLUNT/                                 EDGC 230
C                                                                       EDGC 240
C     SUBROUTINE IECOEF CALCULATE GROUPS OF EDGE QUANTITIES USED IN     EDGC 250
C     THE COEFFICIENTS OF THE GOVERNING PARTIAL DIFFERENTIAL EQUATIONS  EDGC 260
C                                                                       EDGC 270
      PI=DARCOS(-1.DOD)                                                 EDGC 280
      IF (LC.EQ.1) GO TO 10                                             EDGC 290
      B1=XI*DUEWDX/HX(1)/UEW                                            EDGC 300
      B2=XI*DUEWDW/HW(1)/UEW                                            EDGC 310
      XLAMB=UEW**2/TEW                                                  EDGC 320
      ETAX=DUEWDX/DSQRT(2.DO*UEW*XI)-DSQRT(UEW/2.DO/XI**3)              EDGC 330
      ETAW=DUEWDW/DSQRT(2.DO*UEW*XI)                                    EDGC 340
      IF (LC.EQ.2) GO TO 30                                             EDGC 350
      B3=XI*DVEWDX/HX(1)/VEW                                            EDGC 360
      B4=XI*DVEWDW/HW(1)/VEW                                            EDGC 370
      B5=B1                                                             EDGC 380
      B6=B2                                                             EDGC 390
      AL1=VEW/UEW                                                       EDGC 400
      AL2=1.DO                                                          EDGC 410
      AL3S=AL3                                                          EDGC 420
      IF (K.EQ.2) AL3S=0.DO                                             EDGC 430
      AL3=AL1                                                           EDGC 440
      DAL3DX=(AL3-AL3OLD(K))/DXI                                        EDGC 450
      DAL3DW=(AL3-AL3S+AL3OLD(K+1)-AL3OLD(K))/2.DO/DW                   EDGC 460
      AL3OLD(K)=AL3                                                     EDGC 470
      GO TO 40                                                          EDGC 480
   10 CONTINUE                                                          EDGC 490
      AL1=DABS(DVEWDX)/DUEWDX                                           EDGC 500
      B1=1.DO                                                           EDGC 510
      B2=2.DO*AL1                                                       EDGC 520
      B3=1.DO                                                           EDGC 530
      IF (DABS(DVEWDX).LT.1.D-05) B3=2.DO                               EDGC 540
      B4=0.DO                                                           EDGC 550
      IF (K.EQ.1) GO TO 20                                              EDGC 560
      IF (DABS(DVEWDX).GE.1.D-05) B4=DVEWXW/HWX(1)/DVEWDX               EDGC 570
      IF (DABS(DVEWDX).LE.1.D-05) B4=DVWXXW/HWX(1)/DVEWXX               EDGC 580
   20 CONTINUE                                                          EDGC 590
      B5=B3                                                             EDGC 600
      B6=B4                                                             EDGC 610
      AL2=AL1                                                           EDGC 620
      AL3=1.DO                                                          EDGC 630
      AL3OLD(K)=0.DO                                                    EDGC 640
      XLAMB=0.DO                                                        EDGC 650
      GO TO 40                                                          EDGC 660
   30 CONTINUE                                                          EDGC 670
      B3=XI*DVEWXW/DVEWDW                                               EDGC 680
      B4=0.DO                                                           EDGC 690
      B5=B3                                                             EDGC 700
      B6=B4                                                              EDGC 710
      ALB6=X1*DVEWDW/HW(1)/UEW                                           EDGC 720
      AL1=0.D0                                                           EDGC 730
      AL2=0.D0                                                           EDGC 740
      AL3=1.D0                                                           EDGC 750
40    CONTINUE                                                           EDGC 760
      IWT=0                                                              EDGC 770
      IF (IWT.EQ.0) GO TO 50                                             EDGC 780
      WRITE (6,60) B1,B2,B3,B4,B5,B6                                     EDGC 790
      WRITE (6,70) AL1,AL2,AL3,XLAMB,DAL3DX                              EDGC 800
50    CONTINUE                                                           EDGC 810
      RETURN                                                             EDGC 820
C                                                                        EDGC 830
60    FORMAT (1H0,7HB1-6 = ,6E13.6)                                      EDGC 840
70    FORMAT (1H0,8HAL1-3 = ,3E13.6,9H XLAMB = ,E13.6,10H DAL3DX = ,F13.EDGC 850
     16)                                                                 EDGC 860
      END                                                                EDGC 870


      SUBROUTINE EDYVIS                                                  EDYV  10
      IMPLICIT REAL*8(A-H,O-Z)                                           EDYV  20
      REAL*8 KLEB                                                        EDYV  30
      COMMON /DEPVAR/ F(2,101,3),FN(2,101,3),G(2,101,3),GN(2,1Q1,3),T(2,EDYV  40
     1101,3),TN(2,101,3),Y(101),YOL(101)                                 EDYV  50
      COMMON /EDGW/ UEW,VEW,TEW,DUEWDX,DUEWDW,D2UEWW,DVEWDX,DVEWDW,REW   EDYV  60
      COMMON /FRSTRM/ RHOINF,PINF,TFS,UFS,AMUINF,CP,REINF,PR             EDYV  70
      COMMON /GASPRP/ PRANDT(101)                                        EDYV  80
      COMMON /INTEGR/ IE,IM,KEND,KEND2,KLX,K,L,NBLNT1,IND,KPRT,LPRT,KPR,EDYV  90
     1LPR                                                                EDYV 100
      COMMON /SOLPNT/ VW(101),GW(101),TW(101),GWN(101),FWN(101),FW(101),EDYV 110
     1TWN(101),XIW,XW,RW                                                 EDYV 120
      COMMON /SURFAS/ CWALL,CWIND,PEWIND,VWALL,TWALL,XTW(500),TWX(500),XEDYV 130
     1CI(500),CIX(500),TCONW,KCI,KTW                                     EDYV 140
      COMMON /TRBLNT/ ASTAR,AKSTAR,ALAMDA,YSUBL,EVSCTY(101),EDYLAW,EPLUSEDYV 150
     1(101),LAMTRB                                                       EDYV 160
      COMMON /IECOEF/ B1,B2,B3,B4,B5,B6,AL1,AL2,AL3,XLAMB                EDYV 170
      COMMON /HETA/ HX(101),HW(101),HXX(101),HXW(101),HXWW(101),HXN(101)EDYV 180
     1,HXNN(101),HWX(101),HWW(101),HWN(101),HWNN(101)                    EDYV 190
      COMMON /GECOEF/ HXOHXO(101),HWOHWO(101),BHV(101),BHM(101),BHT(101)EDYV 200
     1,XKX(101),XKW(101),BMX(101),BMW(101),XLX(101),XLW(101),BAFF(101),BEDYV 210
     2AGF(101),BAFG(101),BAGG(101),XKXOAL(101)                           EDYV 220
      COMMON /XICORD/ XI,XXI,DXI,XIOLD                                   EDYV 230
      COMMON /ZCOORD/ ETAINF,ETAFAC,ETA(101),DETA(101),ADTEST,KADETA     EDYV 240
      DIMENSION TAU(101), DAMP(101), EPSIN(101), EPSOUT(101), SCALAR(101EDYV 250
     1), VELCTY(101), YPLUS(101)                                         EDYV 260
      DATA SHARP,BLUNT/5HSHARP,5HBLUNT/                                  EDYV 270
      DATA REI,VAN/3HREI,3HVAN/                                          EDYV 280
      DATA CONST/5HCONST/                                                EDYV 290
```

```
C
      IF (XI.EQ.0.0D0) RETURN
C
C     CALCULATE THE PHYSICAL NORMAL DISTANCE PROFILE
C
      EPSVD=DSQRT(1.D0/REINF)
      RETHET=0.0D0
      Y(1)=0.0D0
      YTRANS=DSQRT(XI/2.D0/UEW)
      DO 10 J=2,IE
      Y(J)=Y(J-1)+YTRANS*DETA(J)*EPSVD
      RETHET=RETHET+(FW(J)*(1.0D0-FW(J))+FW(J-1)*(1.0D0-FW(J-1)))*(Y(J)-
     1Y(J-1))/2.0D0
10    CONTINUE
C
C     CALCULATE THE CONSTANT YSUBL
C
      DO 20 N=1,IE
      VELCTY(N)=DSQRT((FW(N)*UEW/HXOHXO(N))**2+(AL1*GW(N)*UEW/HWOHWO(N))
     1**2)/DSQRT((UEW/HXOHXO(N))**2+(VEW/HWOHWO(N))**2)
      NN=N-1
      IF (VELCTY(N).GE.0.99D0) GO TO 30
20    CONTINUE
30    YSUBL=Y(NN)+(Y(NN+1)-Y(NN))*(0.99D0-VELCTY(NN))/(VELCTY(NN+1)-VELC
     1TY(NN))
C
C     CALCULATE THE TOTAL SHEAR FOR USE IN THE VAN DRIEST DAMPING TERM
C     CALCULATE THE SCALAR VELOCITY FUNCTION USED IN THE EDDY VISCOSITY
C
      DO 40 N=1,IE
      DUDY=UFS*DSQRT(2.D0*UEW**3/XI)*HX(1)*(FWN(N)/HX(N)-FW(N)*HXN(N)/HX
     1(N)**2)/EPSVD
      IF (K.EQ.1) DWDY=0.0D0
      IF (K.GT.1) DWDY=UFS*DSQRT(2.D0*UEW**3/XI)*HW(1)*(GWN(N)/HW(N)-GW(
     1N)*HWN(N)/HW(N)**2)/EPSVD
      TAU(N)=AMUINF*DSQRT(DUDY**2+DWDY**2)
      SCALAR(N)=DSQRT(DUDY**2+DWDY**2)
40    CONTINUE
      IF (EDYLAW.EQ.REI) GO TO 70
C
C     CALCULATE THE VAN DRIEST DAMPING TERM FOR THE INNER LAW
C
      DO 50 N=1,IE
      VWPLUS=CWALL*UFS/DSQRT(TAU(1)/RHOINF)
      ASTAR=26.0D0*DEXP(-5.9D0*VWPLUS)
      DAMP(N)=(1.0D0-DEXP(-Y(N)*DSQRT(TAU(N)*RHOINF)/AMUINF/ASTAR))**2
50    CONTINUE
C
C     CALCULATE THE INNER EDDY VISCOSITY, VAN DRIEST EQ.
C
      DO 60 N=1,IE
      EPSIN(N)=RHOINF*AKSTAR**2*Y(N)**2*DAMP(N)*SCALAR(N)/AMUINF
60    CONTINUE
      GO TO 110
C
C     CALCULATE THE INNER EDDY VISCOSITY, REICHARDT EQ.
C
70    CONTINUE
      UPLLST=1.0D0
      UPLUS=0.0D0
      UPL=0.0D0
      EPSIN(1)=0.0D0
      YPLUS(1)=0.0D0
      DO 90 N=2,IE
      VOPLUS=CWALL*UFS/DSQRT(TAU(1)/RHOINF)
      YPLUSA=3.65D0/(VOPLUS+0.344D0)
      YPLUS(N)=Y(N)*DSQRT(TAU(1)*RHOINF)/AMUINF
      EPSIN(N)=AMUINF*0.4D0*(YPLUS(N)-YPLUSA*DTANH(YPLUS(N)/YPLUSA))
      IF (VOPLUS.EQ.0.0D0) GO TO 90
80    CONTINUE
      FACTR=DSQRT(1.0D0+VOPLUS*UPLUS)
      EPSITR=EPSIN(N)*FACTR
      UPL2=UPLUS
      UPLUS=UPL+((1.0D0+UPLUS*VOPLUS)/(1.0D0+EPSITR)+UPLLST)*(YPLUS(N)-Y
     1PLUS(N-1))/2.0D0
      IF (UPL2.EQ.0.0D0) GO TO 80
      IF (DABS((UPLUS-UPL2)/UPL2).GT.0.01D0) GO TO 80
      EPSIN(N)=EPSIN(N)*DSQRT(1.0D0+UPLUS*VOPLUS)
      UPLLST=(1.0D0+UPLUS*VOPLUS)/(1.0D0+EPSIN(N))
      UPL=UPLUS
90    CONTINUE
      DO 100 N=2,IE
100   EPSIN(N)=EPSIN(N)/AMUINF
110   CONTINUE
C
C     CALCULATE THE OUTER EDDY VISCOSITY
C     KLEB IS THE KLEBANOFF INTERMITTANCY FACTOR
C
      DO 120 N=1,IE
      KLEB=1.0D0/(1.0D0+5.5D0*(Y(N)/YSUBL)**6)
      EPSOUT(N)=RHOINF*ALAMDA**2*YSUBL**2*SCALAR(N)*KLEB/AMUINF
120   CONTINUE
C
      OUT=0.0D0
      DO 150 N=1,IE
      IF (OUT.EQ.1.0D0) GO TO 130
      IF (EPSIN(N).GE.EPSOUT(N)) GO TO 130
      EPLUS(N)=EPSIN(N)
      GO TO 140
130   OUT=1.0D0
      EPLUS(N)=EPSOUT(N)
140   CONTINUE
```

```
150   CONTINUE
      DO 160 J=2,IE
160   Y(J)=Y(J)/EPSVD
      RETURN
C
      END



      SUBROUTINE EGPROP (LC)
      IMPLICIT REAL*8(A-H,O-Z)
      REAL*8 NOSE
      COMMON /CONICL/ PE(61),TE(61),UE(61),VE(61),DUEDW(61),DVEDW(61),DP
     1EDWZ(61)
      COMMON /EDGE/ UEDG,TEDG,VEDG,DUEGDX,DUEGDW,D2UEDW,DVEGDX,DVEGDW,RE
     1DG,DRDGDX,ZBDD,XEDG
      COMMON /EDGW/ UEW,VEW,TEW,DUEWDX,DUEWDW,D2UEWW,DVEWDX,DVEWDW,REW
      COMMON /EDG2/ TE2,UE2,VE2,DUE2DX,DVE2DX,DUE2DW,D2UE2W,DVE2DW,RE2,X
     12
      COMMON /FINDIF/ A(101),BB(101),B(101),CC(101),DD(101),D(101),E(101
     1),CRI
      COMMON /GEOM/ ALPHA,NOSE,RNOSE,WLST,X,XX,WX
      COMMON /INTEGR/ IE,IM,KEND,KEND2,KLX,K,L,NBLNT1,IND,KPRT,LPRT,KPR,
     1LPR
      COMMON /OLD/ VOLD(61),CVOLD(61),ROLD(61)
      COMMON /OLDEDG/ R3,UE3,DVE3DW
      COMMON /PDEREF/ UREF
      COMMON /SOLPNT/ VW(101),GW(101),TW(101),GWN(101),FWN(101),FW(101),
     1TWN(101),XIW,DXDXIW,XW,RW
      COMMON /STAG/ PSTAG,TSTAG,PNC,QWSTAG
      COMMON /STGEDG/ DVEGXW,DVEWXW,DVE2XW,DVEGXX,DVEWXX,DVE2XX,DVGXXW,D
     1VWXXW,DV2XXW,DHWOXW,DHXOXW
      COMMON /XICORD/ XI,XXI,DXI,XIOLD
      COMMON /OUTSP/ DUDX,DVDX,DRDX
      COMMON /XSOLVE/ XSTA(100),DXMAX,DX,DXOLD,DX1,NSOLVE
      DATA BLUNT,SHARP/5HBLUNT,5HSHARP/
C
C     BODY EDGE QUANTITIES ARE OBTAINED
C
      IF (L.EQ.1.OR.K.GT.1) GO TO 10
C
C     THE VALUE OF XI IS CALCULATED
C
      DXI=DX
      XI=XIOLD+DXI
      XXI=XIOLD+DX
      XIOLD=XI
10    CONTINUE
      X1=X
```

DATE 05/12/77

```
      XX=X                                                              EGPR 410
      ISNT=1                                                            EGPR 420
      CALL BLUNT2 (ISNT)                                                EGPR 430
      TE2=TEDG                                                          EGPR 440
      RE2=REDG                                                          EGPR 450
      X2=XEDG                                                           EGPR 460
      UE2=UEDG                                                          EGPR 470
      VE2=VEDG                                                          EGPR 480
      DUE2DX=DUEGDX                                                     EGPR 490
      DVE2DX=DVEGDX                                                     EGPR 500
      DUE2DW=DUEGDW                                                     EGPR 510
      D2UE2W=D2UEDW                                                     EGPR 520
      DVE2DW=DVEGDW                                                     EGPR 530
      DVE2XW=DVEGXW                                                     EGPR 540
      IF (LC.NE.1) GO TO 20                                             EGPR 550
      DVE2XX=DVEGXX                                                     EGPR 560
      DV2XXW=DVGXXW                                                     EGPR 570
   20 CONTINUE                                                          EGPR 580
      UEO=UE2                                                           EGPR 590
      IF (CRI.LT.1.0D0) GO TO 30                                        EGPR 600
      TEW=TE2                                                           EGPR 610
      REW=RE2                                                           EGPR 620
      UEW=UE2                                                           EGPR 630
      VEW=VE2                                                           EGPR 640
      DUEWDX=DUE2DX                                                     EGPR 650
      DVEWDX=DVE2DX                                                     EGPR 660
      DUEWDW=DUE2DW                                                     EGPR 670
      D2UEWW=D2UE2W                                                     EGPR 680
      DVEWDW=DVE2DW                                                     EGPR 690
      DVEWXW=DVE2XW                                                     EGPR 700
      IF (LC.NE.1) GO TO 30                                             EGPR 710
      DVEWXX=DVE2XX                                                     EGPR 720
      DVWXXW=DV2XXW                                                     EGPR 730
   30 CONTINUE                                                          EGPR 740
      R2=RE2                                                            EGPR 750
      Z2=ZBOD                                                           EGPR 760
      IF (L.EQ.1) GO TO 40                                              EGPR 770
      ISNT=2                                                            EGPR 780
      X=X-DX/2.0D0                                                      EGPR 790
      XX=X                                                              EGPR 800
      CALL BLUNT2 (ISNT)                                                EGPR 810
      UEO=UEDG                                                          EGPR 820
      RO=REDG                                                           EGPR 830
      IF (CRI.EQ.1.0D0) GO TO 40                                        EGPR 840
      TEW=TEDG                                                          EGPR 850
      REW=REDG                                                          EGPR 860
      UEW=UEDG                                                          EGPR 870
      VEW=VEDG                                                          EGPR 880
      DUEWDX=DUEGDX                                                     EGPR 890
      DVEWDX=DVEGDX                                                     EGPR 900
      DUEWDW=DUEGDW                                                     EGPR 910
      D2UEWW=D2UEDW
      DVEWDW=DVEGDW
      DVEWXW=DVEGXW
      IF (LC.NE.1) GO TO 40
      DVEWXX=DVEGXX
      DVWXXW=DVGXXW
40    CONTINUE
      Z0=ZBOD
      ZBOD=Z2
      X=X1
C
C     PROPERTIES AT THE BOUNDARY LAYER EDGE ARE CALCULATED
C
      IF (NOSE.EQ.BLUNT.AND.L.EQ.1) PNC=DSQRT(2.0D0*DUE2DX)
      DUDX=DUE2DX
      DVDX=DVE2DX
      DRDX=DRE2DX
      IF (L.EQ.1) GO TO 50
      XW=X-DX*(1.0D0-CRI)
      RW=RE2
      IF (CRI.LT.1.0D0) RW=REW
      XIW=XI
      IF (CRI.EQ.1.0D0) GO TO 50
      XIW=XXI
50    CALL EDGCOF (LC)
      CALL GEOCOF
      RETURN
      END


      SUBROUTINE ENERGY (LC)
      IMPLICIT REAL*8(A-H,O-Z)
      REAL*8 NOSE
      COMMON /FRSTRM/ RHOINF,PINF,TFS,UFS,AMUINF,CP,REINF,PR
      COMMON /GASPRP/ PRANDT(101)
      COMMON /GEOM/ ALPHA,NOSE,RNOSE,WLST,X,XX,WX
      COMMON /IECOEF/ B1,B2,B3,B4,B5,B6,AL1,AL2,AL3,XLAMB
      COMMON /GECOEF/ HXOHXO(101),HWOHWO(101),BHV(101),BHM(101),BHT(101)
     1,XKX(101),XKW(101),BMX(010),BMW(101),XLX(101),XLW(101),BAFF(101),B
     2AGF(101),BAFG(101),BAGG(101),XKXOAL(101)
      COMMON /INTEGR/ IE,IM,KEND,KEND2,KLX,K,L,NBLNT1,IND,KPRT,LPRT,KPR,
     1LPR
      COMMON /PDECOF/ A0(101),A1(101),A2(101),A3(101),A4(101),A5(101)
      COMMON /SOLPNT/ VW(101),GW(101),TW(101),GWN(101),FWN(101),FW(101),
     1TWN(101),XIW,XW,RW
      COMMON /TRANSN/ KTRANS,KONSET,XIF,CHI2(101),CHIMAX,XBAR
      COMMON /TRBLNT/ ASTAR,AKSTAR,ALAMDA,YSUBL,EVSCTY(101),EDYLAW,EPLUS
     1(101),LAMTRB
```

```
      COMMON /XICORD/ XI,XXI,DXI,XIOLD                                  ENER 190
      COMMON /ZCOORD/ ETAINF,ETAFAC,ETA(101),DETA(101),ADTEST,KADETA    ENER 200
      COMMON /NCOORD/ XN(101),ETAX,ETAW                                 ENER 210
      COMMON /EDGW/ UEW,VEW,TEW,DUEWDX,DUEWDW,D2UEWW,DVEWDX,DVEWDW,REW  ENER 220
      COMMON /HETA/ HX(101),HW(101),HXX(101),HXW(101),HXWW(101),HXN(101)ENER 230
     1,HXNN(101),HWX(010),HWW(101),HWN(101),HWNN(010)                   ENER 240
      COMMON /INTGRL/ XIFNX(101),XIFNW(101),XIFNXX(010),XIFNWW(101)     ENER 250
      COMMON /STGEDG/ DVEGXW,DVEWXW,DVE2XW,DVEGXX,DVEWXX,DVE2XX,DVGXXW,DENER 260
     1VWXXW,DV2XXW,DHWOXW,DHXOXW                                        ENER 270
      DIMENSION ROMU1(101), ROMU1N(101)                                 ENER 280
      DATA SHARP,BLUNT/5HSHARP,5HBLUNT/                                 ENER 290
C                                                                       ENER 300
C     SUBROUTINE ENERGY SETS UP THE COEFFICIENTS OF THE PARTIAL         ENER 310
C     DIFFERENTIAL ENERGY EQUATION                                      ENER 320
C                                                                       ENER 330
      DO 10 J=1,IE                                                      ENER 340
      ROMU1(J)=1.DO/PR*(1.0DO+XIF*EPLUS(J)*PR/PRANDT(J))                ENER 350
   10 CONTINUE                                                          ENER 360
      CALL DERIV (ROMU1,ETA,IE,1,ROMU1N)                                ENER 370
      DO 40 J=1,IE                                                      ENER 380
      A0(J)=ROMU1(J)                                                    ENER 390
      A1(J)=-ROMU1N(J)*BHT(J)-VW(J)                                     ENER 400
      A2(J)=0.0DO                                                       ENER 410
      A3(J)=((HX(1)*FWN(J))**2+4.DO*(HX(1)*HXN(J)/HX(J)**2)**2)*XLAMB   ENER 420
      IF (LC.EQ.1) GO TO 20                                             ENER 430
      IF (LC.EQ.2) GO TO 30                                             ENER 440
      A3(J)=A3(J)+((HW(1)*AL2*GW(J))**2+4.DO*(HW(1)*HWN(J)*AL2/HW(J)**2)ENER 450
     1**2)*XLAMB                                                        ENER 460
      A4(J)=-HX(1)*XI*FW(J)/HX(J)**2                                    ENER 470
      A5(J)=-HW(1)*XI*AL2*GW(J)/HW(J)**2                                ENER 480
      GO TO 40                                                          ENER 490
   20 CONTINUE                                                          ENER 500
      A4(J)=0.DO                                                        ENER 510
      A5(J)=-AL1*GW(J)/2.DO/HWX(1)                                      ENER 520
      GO TO 40                                                          ENER 530
   30 CONTINUE                                                          ENER 540
      A4(J)=-XI*HX(1)*FW(J)/2.DO/HX(J)**2                               ENER 550
      A5(J)=0.DO                                                        ENER 560
   40 CONTINUE                                                          ENER 570
      RETURN                                                            ENER 580
      END                                                               ENER 590



      SUBROUTINE FD3 (X,X1,X2,X3,F1,F2,F3,FX)                           FD30  10
      IMPLICIT REAL*8 (A-H,O-Z)                                         FD30  20
C                                                                       FD30  30
C     SUBROUTINE FD3 CALCULATES THE FIRST DERIVATIVE-FX-CORRESPONDING   FD30  40
C     TO POINT X USING 3 POINT LAGRANGIAN DIFFERENTIATION FORMULA.      FD30  50
```

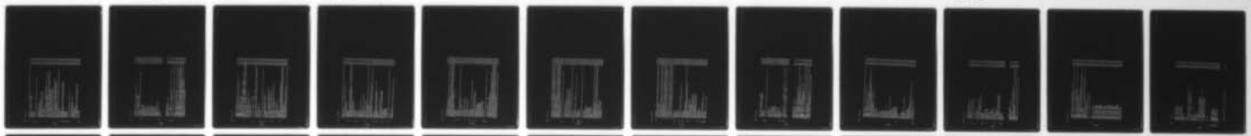
DATE 05/12/77

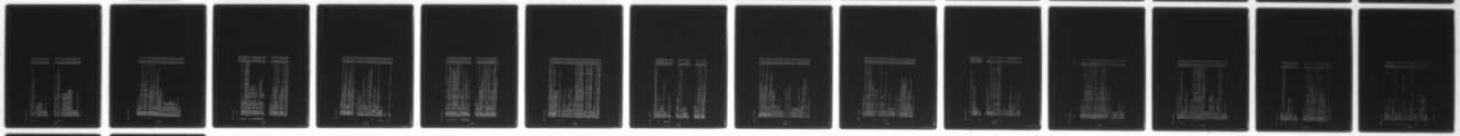
```
C                                                                       FD30  60
C     ASSUMES X1 .LE. X .LE. X3.                                        FD30  70
C                                                                       FD30  80
      A1=2.0*X-X2-X3                                                    FD30  90
      A2=2.0*X-X1-X3                                                    FD30 100
      A3=2.0*X-X1-X2                                                    FD30 110
      D1=(X1-X2)*(X1-X3)                                                FD30 120
      D2=(X2-X1)*(X2-X3)                                                FD30 130
      D3=(X3-X1)*(X3-X2)                                                FD30 140
      C1=A1/D1                                                          FD30 150
      C2=A2/D2                                                          FD30 160
      C3=A3/D3                                                          FD30 170
      FX=C1*F1+C2*F2+C3*F3                                              FD30 180
      RETURN                                                            FD30 190
      END                                                               FD30 200


      SUBROUTINE FD5 (X,X1,X2,X3,X4,X5,F1,F2,F3,F4,F5,FX)               FD50  10
      IMPLICIT REAL*8 (A-H,O-Z)                                         FD50  20
C                                                                       FD50  30
C     SUBROUTINE FD5 CALCULATES THE FIRST DERIVATIVE-FX-CORRESPONDING   FD50  40
C     TO POINT X USING 5 POINT LAGRANGIAN DIFFERENTIATION FORMULA       FD50  50
C                                                                       FD50  60
C     ASSUMES X1 .LE. X .LE. X5.                                        FD50  70
C                                                                       FD50  80
      A1=(X-X4)*(X-X5)*(2.0*X-X2-X3)+(X-X2)*(X-X3)*(2.0*X-X4-X5)        FD50  90
      A2=(X-X4)*(X-X5)*(2.0*X-X1-X3)+(X-X1)*(X-X3)*(2.0*X-X4-X5)        FD50 100
      A3=(X-X4)*(X-X5)*(2.0*X-X1-X2)+(X-X1)*(X-X2)*(2.0*X-X4-X5)        FD50 110
      A4=(X-X3)*(X-X5)*(2.0*X-X1-X2)+(X-X1)*(X-X2)*(2.0*X-X3-X5)        FD50 120
      A5=(X-X3)*(X-X4)*(2.0*X-X1-X2)+(X-X1)*(X-X2)*(2.0*X-X3-X4)        FD50 130
      D1=(X1-X2)*(X1-X3)*(X1-X4)*(X1-X5)                                FD50 140
      D2=(X2-X1)*(X2-X3)*(X2-X4)*(X2-X5)                                FD50 150
      D3=(X3-X1)*(X3-X2)*(X3-X4)*(X3-X5)                                FD50 160
      D4=(X4-X1)*(X4-X2)*(X4-X3)*(X4-X5)                                FD50 170
      D5=(X5-X1)*(X5-X2)*(X5-X3)*(X5-X4)                                FD50 180
      C1=A1/D1                                                          FD50 190
      C2=A2/D2                                                          FD50 200
      C3=A3/D3                                                          FD50 210
      C4=A4/D4                                                          FD50 220
      C5=A5/D5                                                          FD50 230
      FX=C1*F1+C2*F2+C3*F3+C4*F4+C5*F5                                  FD50 240
      RETURN                                                            FD50 250
      END                                                               FD50 260
      SUBROUTINE GEOCOF                                                 GEOC  10
      IMPLICIT REAL*8(A-H,O-Z)                                          GEOC  20
C                                                                       GEOC  30
C      THIS SUBROUTINE CALCULATES THE GEOMETRIC COEFFICIENTS OF THE     GEOC  40
C      GOVERNING EQUATIONS                                              GEOC  50
C                                                                       GEOC  60
      COMMON /INTEGR/ IE,IM,KEND,KEND2,KLX,K,L,NBLNT1,IND,KPRT,LPRT,KPR,GEOC  70
     1LPR                                                               GEOC  80
      COMMON /XICORD/ XI,XXI,DXI,XIOLD                                  GEOC  90
      COMMON /EDGW/ UEW,VEW,TEW,DUEWDX,DUEWDW,D2UEWW,DVEWDX,DVEWDW,REW  GEOC 100
      COMMON /NCOORD/ XN(101),ETAX,ETAW                                 GEOC 110
      COMMON /HETA/ HX(101),HW(101),HXX(101),HXW(101),HXWW(101),HXN(101)GEOC 120
     1,HXNN(101),HWX(101),HWW(101),HWN(101),HWNN(101)                   GEOC 130
      COMMON /GECOEF/ HXOHXO(101),HWOHWO(101),BHV(101),BHM(101),BHT(101)GEOC 140
     1,XKX(101),XKW(101),BMX(101),BMW(101),XLX(101),XLW(101),BAFF(101),BGEOC 150
     2AGF(101),BAFG(101),BAGG(101),XKXOAL(101)                          GEOC 160
      COMMON /STGEDG/ DVEGXW,DVEWXW,DVE2XW,DVEGXX,DVEWXX,DVE2XX,DVGXXW,DGEOC 170
     1VWXXW,DV2XXW,DHWOXW,DHXOXW                                        GEOC 180
      COMMON /WSOLVE/ DW,W                                              GEOC 190
      COMMON /STGSPC/ HXXOLD(101)                                       GEOC 200
      COMMON /TVCURV/ TVC                                               GEOC 210
      COMMON /EDGE/ B1,B2,B3,B4,B5,B6,AL1,AL2,AL3,XLAMB                 GEOC 220
      DATA YES/3HYES/                                                   GEOC 230
      PI=DARCOS(-1.D0)                                                  GEOC 240
      IF (TVC.NE.YES) GO TO 80                                          GEOC 250
C                                                                       GEOC 260
C     CALCULATE COEFFICIENTS WITH TVC                                   GEOC 270
C                                                                       GEOC 280
C     CALCULATE COEFFICIENTS AT STAGNATION POINT                        GEOC 290
C                                                                       GEOC 300
      IF (L.NE.1) GO TO 40                                              GEOC 310
      DO 30 I=1,IE                                                      GEOC 320
      HXOHXO(I)=HX(I)/HX(1)                                             GEOC 330
      HWOHWO(I)=1.D0                                                    GEOC 340
      BHV(I)=HXN(I)/HX(I)                                               GEOC 350
      BHM(I)=BHV(I)                                                     GEOC 360
      BHT(I)=BHM(I)                                                     GEOC 370
      XKX(I)=HXW(I)/(HX(I)*HWX(I))                                      GEOC 380
      XKW(I)=1.D0/HX(I)                                                 GEOC 390
      BMX(I)=HXNN(I)/HX(I)-(HXN(I)/HX(I))**2                            GEOC 400
      BMW(I)=0.D0                                                       GEOC 410
      XLX(I)=0.D0                                                       GEOC 420
      XLW(I)=0.D0                                                       GEOC 430
      IF (DABS(DVEWDX).LT.1.D-05) GO TO 10                              GEOC 440
      XKXOAL(I)=HXW(I)/HX(I)/HWX(I)/AL1                                 GEOC 450
      GO TO 20                                                          GEOC 460
   10 CONTINUE                                                          GEOC 470
      XKXOAL(I)=3.D0                                                    GEOC 480
   20 CONTINUE                                                          GEOC 490
      BAFF(I)=0.D0                                                      GEOC 500
      BAGF(I)=0.D0                                                      GEOC 510
```

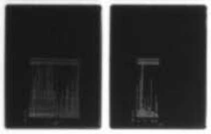
AD-A051 971 VIRGINIA POLYTECHNIC INST AND STATE UNIV BLACKSBURG --ETC F/G 20/4
THREE-DIMENSIONAL INCOMPRESSIBLE BOUNDARY LAYERS ON BLUNT BODIE--ETC(U)
MAY 77 D L DWOYER, C H LEWIS, P R GOGINENI

UNCLASSIFIED VPI/SU-AERO-063-PT-2 NL

3 OF 3
ADA
051971







END
DATE
FILMED
4 -78
DDC

```
      BAFG(I)=((DUEWDX/HXOHXO(I))**2*HXN(I)+DSQRT(DUEWDX**3/2.DO)/HXOHXO
     1(I)**3)/DSQRT(2.DO*DUEWDX**3)
      BAGG(I)=DVEWDX/2.DO/DUEWDX**2
30    CONTINUE
      RETURN
40    CONTINUE
C
C     CALCULATE GEOMETRIC COEFFICIENTS FOR A GENERAL POINT
C
      DO 70 I=1,IE
      HXOHXO(I)=HX(I)/HX(1)
      HWOHWO(I)=HW(I)/HW(1)
      BHV(I)=HWN(I)/HW(I)+HXN(I)/HX(I)
      BHM(I)=HXN(I)/HX(I)-HWN(I)/HW(I)
      BHT(I)=BHV(I)
      XKX(I)=0.DO
      IF (K.NE.1) XKX(I)=XI*HXW(I)/HX(I)/HW(I)
      IF (K.EQ.1) XKXOAL(I)=XI*HXWW(I)*UEW/HX(I)/HW(I)/DVEWDW
      XKW(I)=XI*HWX(I)/HX(I)/HW(I)
      BMX(I)=(HWN(I)*HXN(I)/HW(I)-HXN(I)**2/HX(I)+HXNN(I))/HX(I)
      BMW(I)=(HWN(I)*HXN(I)/HX(I)-HWN(I)**2/HW(I)+HWNN(I))/HW(I)
      XLX(I)=XI*HXX(I)/HXOHXO(I)/HX(I)**2
      XLW(I)=XI*HWW(I)/HWOHWO(I)/HW(I)**2
      BAFFC=UEW**2*HXN(I)/HXOHXO(I)**2/HX(I)
      BAGFC=0.DO
      IF (K.NE.1) BAGFC=UEW**2*HWN(I)/HWOHWO(I)**2/HW(I)
      BAFF(I)=XI*BAFFC/2.DO/HX(1)/UEW**2*ETAX*XN(I)
      BAGF(I)=XI*BAGFC/2.DO/HX(I)/UEW**2*ETAX*XN(I)
      IF (K.NE.1) GO TO 50
      BAFG(I)=BAFFC*DSQRT(XI/2.DO/UEW**3)/HW(1)*D2UEWW/DVEWDW*XN(I)
      BAGG(I)=0.DO
      GO TO 60
50    CONTINUE
      BAFG(I)=XI*BAFFC/2.DO/HW(1)/UEW**2*ETAW*XN(I)
      BAGG(I)=XI*BAGFC/2.DO/HW(1)/UEW**2*ETAW*XN(I)
60    CONTINUE
70    CONTINUE
      RETURN
80    CONTINUE
C
C     CALCULATE COEFFICIENTS WITHOUT TVC
C
      IF (L.NE.1) GO TO 90
      XKX(1)=HXW(1)/(HX(1)*HWX(1))
      XKW(1)=1.DO/HX(1)
      XLX(1)=0.DO
      XLW(1)=0.DO
      IF (DABS(DVEWDX).GT.1.D-05) XKXOAL(1)=0.DO
      IF (DABS(DVEWDX).LE.1.D-05) XKXOAL(1)=3.DO
      GO TO 100
90    CONTINUE
```

```
      XKX(1)=0.D0                                                       GEOC1030
      IF (K.NE.1) XKX(1)=XI*HXW(1)/HX(1)/HW(1)                          GEOC1040
      XKW(1)=XI*HWX(1)/HX(1)/HW(1)                                      GEOC1050
      XLX(1)=XI*HXX(1)/HX(1)**2                                         GEOC1060
      XLW(1)=XI*HWW(1)/HW(1)**2                                         GEOC1070
      IF (K.EQ.1) XKXOAL(1)=XI*HXWW(1)*UEW/HX(1)/HW(1)/DVEWDW           GEOC1080
100   CONTINUE                                                          GEOC1090
      DO 110 I=1,IE                                                     GEOC1100
      HXOHXO(I)=1.D0                                                    GEOC1110
      HWOHWO(I)=1.D0                                                    GEOC1120
      BHV(I)=0.D0                                                       GEOC1130
      BHM(I)=0.D0                                                       GEOC1140
      BHT(I)=0.D0                                                       GEOC1150
      XKX(I)=XKX(1)                                                     GEOC1160
      XKW(I)=XKW(1)                                                     GEOC1170
      BMX(I)=0.D0                                                       GEOC1180
      BMW(I)=0.D0                                                       GEOC1190
      XLX(I)=XLX(1)                                                     GEOC1200
      XLW(I)=XLW(1)                                                     GEOC1210
      BAFF(I)=0.D0                                                      GEOC1220
      BAGF(I)=0.D0                                                      GEOC1230
      BAFG(I)=0.D0                                                      GEOC1240
      BAGG(I)=0.D0                                                      GEOC1250
      XKXOAL(I)=XKXOAL(1)                                               GEOC1260
110   CONTINUE                                                          GEOC1270
      RETURN                                                            GEOC1280
      END                                                               GEOC1290


      SUBROUTINE HOFN                                                   HOFN  10
      IMPLICIT REAL*8(A-H,O-Z)                                          HOFN  20
      REAL*8 NOSE                                                       HOFN  30
C                                                                       HOFN  40
C     THIS SUBROUTINE CALCULATES THE NORMAL VARIATION OF THE METRIC     HOFN  50
C     COEFFICIENTS AND THEIR DERIVATIVES                                HOFN  60
C                                                                       HOFN  70
      COMMON /INTEGR/ IE,IM,KEND,KEND2,KLX,K,L,NBLNT1,IND,KPRT,LPRT,KPR,HOFN  80
     1LPR                                                               HOFN  90
      COMMON /XICORD/ XI,XXI,DXI,XIOLD                                  HOFN 100
      COMMON /EDGE/ UEDG,TEDG,VEDG,DUEGDX,DUEGDW,D2UEGW,DVEGDX,DVEGDW,REHOFN 110
     1DG,DRGDX,ZBOD,XEDG                                                HOFN 120
      COMMON /ZCOORD/ ETAINF,ETAFAC,ETA(101),DETA(101),ADTEST,KADETA    HOFN 130
      COMMON /HRWALL/ RO,HXO,HWO,DRODX,DRODW,D2RDXX,D2RDWW,D2RDXW,D3RXXXHOFN 140
     1,D3RWXX,D3RWWX,D3RWWW,DRWWXX,DRWWWX,DHXODX,DHXODW,DHXDWW,DHWODX,DHHOFN 150
     2WODW,XO,DXODX,DXODW,D2XODX,D2XODW,D2XDWX,D3XXWW                   HOFN 160
      COMMON /HETA/ HX(101),HW(101),HXX(101),HXW(101),HXWW(101),HXN(101)HOFN 170
     1,HXNN(101),HWX(101),HWW(101),HWN(101),HWNN(101)                   HOFN 180
      COMMON /NCOORD/ XN(101),ETAX,ETAW                                 HOFN 190
      COMMON /WSOLVE/ DW,W
      COMMON /FRSTRM/ RHOINF,PINF,TFS,UFS,AMUINF,CP,REINF,PR
      COMMON /GEOM/ ALPHA,NOSE,RNOSE,WLST,X,XX,WX
      COMMON /TVCURV/ TVC
      COMMON /STGSPC/ HXXOLD(101)
      COMMON /RAD/ RXS(2,101,3)
      DIMENSION TXXS(61), TWS(61), TWXS(61), HTXS(101)
      DIMENSION RT(101), RTW(101), RTWW(101), RTX(101), RTXX(101), RTWX(
     1101), PHT(101), PHTW(101), PHTWW(101), PHTX(101), PHTXX(101), PHTW
     2X(101), XT(101), XTW(101), XTWW(101), XTX(101), XTXX(101), XTWX(10
     31), RTWWX(101), XTXWW(101)
      DATA YES/3HYES/
      DATA BLUNT,SHARP/5HBLUNT,5HSHARP/
      EPSVD=1.D0/DSQRT(REINF)
      PI=DARCOS(-1.D0)
      IF (TVC.NE.YES) GO TO 310
C
C     CALCULATE NORMAL COORDINATE
C
      IF (L.EQ.1.AND.NOSE.EQ.BLUNT) YTRANS=1.D0/DSQRT(2.D0*DUEGDX)*EPSVD
      IF (L.NE.1.OR.NOSE.EQ.SHARP) YTRANS=DSQRT(XI/(2.D0*UEDG))*EPSVD
      XN(1)=0.D0
      DO 10 I=2,IE
      XN(I)=YTRANS*ETA(I)
   10 CONTINUE
C
C     CALCULATE BODY SLOPES AND DERIVATIVES IN X AND PHI DIRECTIONS
C
      IF (L.EQ.1.AND.K.GT.1) GO TO 20
      TX=DATAN(DABS(DRODX/DXODX))
   20 CONTINUE
      TW=0.D0
      IF (L.GT.1) TW=DATAN(DRODW/2.D0/PI/RO)
      AX=1.D0/(1.D0+(DRODX/DXODX)**2)
      IF (L.EQ.1) GO TO 30
      AW=1.D0/(1.D0+.25D0*(DRODW/PI/RO)**2)
   30 CONTINUE
      TXX=(-D2XODX*DRODX/DXODX**2+D2RDXX/DXODX)*AX
      IF (L.EQ.1) TXXS(K)=TXX
      TXXX=0.D0
      IF (L.EQ.1) GO TO 40
      TXXX=(TXX-TXXS(K))/DXI
      TXXS(K)=TXX
   40 CONTINUE
      TXW=(-D2XDWX*DRODX/DXODX**2+D2RDXW/DXODX)*AX
      TXWW=-TXW*(-2.D0*D2XDWX*DRODX**2/DXODX**3+2.D0*DRODX*D2RDXW/DXODX*
     1*2)*AX+(2.D0*D2XDWX**2*DRODX/DXODX**3-2.D0*D2XDWX*D2RDXW/DXODX**2+
     2D3RWWX/DXODX)*AX
      IF (L.EQ.1) GO TO 70
      TWW=(D2RDWW/RO-(DRODW/RO)**2)*AW/2.D0/PI
      TWWW=-TWW*(DRODW*D2RDWW/2.D0*PI**2/RO**2-(DRODW/RO)**3/2.D0/PI**2)
     1*AW+(D3RWWW/RO-2.DO*D2RDWW*DRODW/RO**2+2.DO*DRODW**2*D2RDWW/RO**3)HOFN 710
     2*AW/2.DO/PI                                                       HOFN 720
      TWX=(D2RDXW/RO-DRODW*DRODX/RO**2)*AW/2.DO/PI                      HOFN 730
      IF (L.EQ.1) GO TO 50                                              HOFN 740
      TWX=(TW-TWS(K))/DXI                                               HOFN 750
      TWS(K)=TW                                                         HOFN 760
50    CONTINUE                                                          HOFN 770
      TWXX=-TWX*(DRODW*D2RDXW/2.DO*PI**2/RO**2-DRODW**2*DRODX/RO**3/2.DOHOFN 780
     1/PI**2)*AW+(D3RWXX/RO-2.DO*D2RDXW*DRODX/RO**2-DRODW*D2RDXW/RO**2+DHOFN 790
     2RODW*DRODX**2*2.DO/RO**3)*AW/2.DO/PI                              HOFN 800
      IF (L.EQ.1) GO TO 60                                              HOFN 810
      TWXX=(TWX-TWXS(K))/DXI                                            HOFN 820
      TWXS(K)=TWX                                                       HOFN 830
60    CONTINUE                                                          HOFN 840
      GO TO 80                                                          HOFN 850
70    CONTINUE                                                          HOFN 860
      TWW=0.DO                                                          HOFN 870
      TWWW=0.DO                                                         HOFN 880
      TWX=D3RWXX/DRODX/2.DO/PI                                          HOFN 890
      TWS(K)=TW                                                         HOFN 900
      TWXS(K)=TWX                                                       HOFN 910
      TWXX=-D3RWXX*D2RDXX/DRODX**2/PI                                   HOFN 920
80    CONTINUE                                                          HOFN 930
      IF (K.GT.1) GO TO 100                                             HOFN 940
      TXWX=D3RWXX*AX/DXODX                                              HOFN 950
      TXWWX=-2.DO*DRODX*D3RWWX/DXODX**2*AX**2+(-D2XODX*D3RWWX/DXODX**2+DHOFN 960
     1RWWXX/DXODX)*AX                                                   HOFN 970
      IF (L.EQ.1) GO TO 90                                              HOFN 980
      TWWX=(D3RWWX/RO-D2RDWW*DRODX/RO**2)/2.DO/PI                       HOFN 990
      TWWWX=((D2RDWW/RO)**3/2.DO/PI**2+DRWWWX/RO-D3RWWX*DRODX/RO**2)/2.DHOFN1000
     1O/PI                                                              HOFN1010
      GO TO 100                                                         HOFN1020
90    CONTINUE                                                          HOFN1030
      TWWX=(DRWWXX/DRODX-D3RWXX*D2RDXX/DRODX)/2.DO/PI                   HOFN1040
      TWWWX=-DRWWXX*D2RDXX/DRODX**2/PI                                  HOFN1050
100   CONTINUE                                                          HOFN1060
      IWT=0                                                             HOFN1070
      IF (IWT.EQ.0) GO TO 110                                           HOFN1080
      WRITE (6,330) TX,TW,TXX,TXW,TWX,TWW                               HOFN1090
      WRITE (6,330) TXXX,TXWW,TWXX,TWWW                                 HOFN1100
110   CONTINUE                                                          HOFN1110
      STX=DSIN(TX)                                                      HOFN1120
      CTX=DCOS(TX)                                                      HOFN1130
      STW=DSIN(TW)                                                      HOFN1140
      CTW=DCOS(TW)                                                      HOFN1150
C                                                                       HOFN1160
C     CALCULATE RADIUS AND DERIVATIVES OF RADIUS FOR PROFILE POINTS     HOFN1170
C                                                                       HOFN1180
      IINC=5                                                            HOFN1190
      DO 210 I=2,IE                                                     HOFN1200
      R=DSQRT(RO**2+(XN(I)*CTX)**2+2.DO*RO*XN(I)*CTX*CTW)               HOFN1210
      RT(I)=R-RO
      IF (L.EQ.1) GO TO 120
      RW=(RO*DRODW-XN(I)**2*CTX*STX*TXW+DRODW*XN(I)*CTX*CTW-RO*XN(I)*(ST
     1X*CTW*TXW+CTX*STW*TWW))/R
      RWW=(-RW**2+DRODW**2+RO*D2RDWW+XN(I)**2*((STX*TXW)**2-(CTX*TXW)**2
     1+CTX*STX*TXWW)+D2RDWW*XN(I)*CTX*CTW-DRODW*XN(I)*(STX*CTW*TXW+CTX*S
     2TW*TWW)-RO*XN(I)*(CTX*CTW*TXW**2-2.DO*STX*STW*TXW*TWW+CTX*STW*TWW*
     3*2+STX*CTW*TXWW+CTX*STW*TWWW))/R
      GO TO 130
120   CONTINUE
      RW=0.DO
      RWW=0.DO
130   CONTINUE
      RX=(RO*DRODX-XN(I)**2*CTX*STX*TXX+DRODX*XN(I)*CTX*CTW-RO*XN(I)*(ST
     1X*CTW*TXX+CTX*STW*TWX))/R
      RXS(2,I,2)=RX
      RXX=0.DO
      IF (L.EQ.1) GO TO 140
      RXX=(RX-RXS(1,I,2))/DXI
140   CONTINUE
      IF (K.EQ.1) GO TO 150
      RWX=(RX-RXS(2,I,1))/DW
      GO TO 160
150   CONTINUE
      RWX=0.DO
      RWWX=D3RWWX-XN(I)*CTX*TXX*TXWW-XN(I)*STX*TXWWX
160   CONTINUE
      IF (IINC.NE.5000) GO TO 170
      WRITE (6,340) R,RX,RW,RXX,RWX,RWW
170   CONTINUE
      RTW(I)=RW-DRODW
      RTWW(I)=RWW-D2RDWW
      RTX(I)=RX-DRODX
      RTXX(I)=RXX-D2RDXX
      RTWX(I)=RWX-D2RDXW
      IF (K.NE.1) GO TO 180
      RTWWX(I)=RWWX-D3RWWX
180   CONTINUE
C
C     CALCULATE MERIDIONAL ANGLE AND DERIVATIVES FOR PROFILE POINTS
C
      XNHAT=XN(I)*CTX
      PHT(I)=DARSIN(XNHAT*STW/R)
      D=DSQRT(1.DO-(XNHAT*STW/R)**2)
      PHTW(I)=XNHAT*(CTW*TWW-STW*RW/R)/R/D-XN(I)*STX*STW*TXW/R/D
      PHTWW(I)=-PHTW(I)*XN(I)**2*(CTX**2*STW*CTW*TWW-STX*CTX*STW**2*TXW-
     1TX**2*STW**2*RW/R)/R**2/D**2+XN(I)*(-STX*CTW*TWW*TXW-CTX*STW*TWW**
     22+CTX*CTW*TWWW-CTX*CTW*TWW*RW/R-CTX*STW*TXW**2-STX*CTW*TXW*TWW-STX
     3*STW*TXWW+2.DO*STX*STW*RW*TXW/R-CTX*CTW*TWW*RW*R+2.DO*CTX*STW*RW**
     42/R**2-CTX*STW*RWW/R)/R/D
      PHTX(I)=XNHAT*(CTW*TWX-STW*RX/R)/R/D-XN(I)*STX*STW*TXX/R/D
      PHTXX(I)=-PHTX(I)*XN(I)**2*(CTX**2*STW*CTW*TWX-STX*CTX*STW**2*TXX-
     1CTX**2*STW**2*RX/R)/R**2/D**2+XN(I)*(-STX*CTW*TWX*TXX-CTX*STW*TWX*
     2*2+CTX*CTW*TWXX-CTX*CTW*TWX*RX/R-CTX*STW*TXX**2-STX*CTW*TXX*TWX-ST
     3X*STW*TXXX+2.DO*STX*STW*TXX*RX/R-CTX*CTW*TWX*RX/R+2.DO*CTX*STW*RW*
     4*2/R**2-CTX*STW*RWW/R)/R/D
      PHTWX(I)=-PHTW(I)*XN(I)**2*(CTX**2*STW*CTW*TWX-STX*CTX*STW**2*TXX-
     1CTX**2*STW**2*RX/R)/R**2/D**2+XN(I)*(-STX*CTW*TWX*TXX-CTX*STW*TWX*
     2TWW+CTX*CTW*TWWX-CTX*CTW*TWX*RX/R-CTX*STW*TXX*TXW-STX*CTW*TXX*TWX-
     3STX*STW*TXWX+2.DO*STX*STW*TXX*RX/R-CTX*CTW*TWX*RX/R+CTX*STW*RX*RW/
     4R**2-CTX*STW*RWX/R)/R/D
C
C     CALCULATE AXIAL DISTANCE AND DERIVATIVES FOR PROFILE POINTS
C
      XT(I)=-XN(I)*STX
      XTW(I)=-XN(I)*CTX*TXW
      XTWW(I)=XN(I)*(STX*TXW**2+CTX*TXWW)
      XTX(I)=-XN(I)*CTX*TXX
      XTXX(I)=XN(I)*(STX*TXX**2+CTX*TXXX)
      XTWX(I)=-XN(I)*(STX*TXX*TXW+CTX*TXWX)
      IF (K.NE.1) GO TO 190
      XTXWW(I)=-XN(I)*(CTX*TXX*TXW**2+2.DO*STX*TXW*TXXW+STX*TXWW-CTX*TXW
     1WX)
190   CONTINUE
      IF (IINC.NE.5000) GO TO 200
      WRITE (6,340) PHT(I),PHTX(I),PHTW(I),XT(I),XTX(I),XTW(I)
      IINC=0
200   CONTINUE
      IINC=IINC+1
210   CONTINUE
C
C     CALCULATE METRIC COEFFICIENTS AND DERIVATIVES FOR PROFILE POINTS
C
      HX(1)=HXO
      HW(1)=HWO
      HXXOLD(1)=HXX(1)
      HXX(1)=DHXODX
      HXW(1)=DHXODW
      HWX(1)=DHWODX
      HWW(1)=DHWODW
      HXWW(1)=DHXDWW
      IINC=5
      DO 300 I=2,IE
      HTX=RTX(I)*(2.DO*DRODX+RTX(I))+((RO+RT(I))*PHTX(I))**2+2.DO*XTX(I)
     1*(2.DO*DXODX+XTX(I))
      HTXW=RTWX(I)*DRODX+RTX(I)*(D2RDXW+RTWX(I))+(RO+RT(I))*(DRODW+RTW(I
     1))*PHTX(I)**2+(RO+RT(I))**2*PHTX(I)*PHTWX(I)+XTWX(I)*DXODX+XTX(I)*
     2(D2XDWX+XTWX(I))
      IF (K.EQ.1) HTXS(I)=HTX
      IF (K.EQ.1) GO TO 220
      HTXW=(HTX-HTXS(I))/DW
      HTXS(I)=HTX
```

```
220    CONTINUE                                                                  HOFN2240
       HTXX=RTXX(I)*DRODX+PTX(I)*(D2RDXX+RTXX(I))+(RO+RT(I))*(DRODX+RTX(IHOFN2250
      1))*PHTX(I)**2+(RO+RT(I))**2*PHTX(I)*PHTXX(I)+XTXX(I)*DXODX+XTX(I)*HOFN2260
      2(D2XODX+XTXX(I))                                                          HOFN2270
       IF (L.EQ.1) GO TO 230                                                     HOFN2280
       HTW=4.DO*PI**2*(RT(I)**2+2.DO*RT(I)*RO)+RTW(I)**2+2.DO*DRODW*RTW(IHOFN2290
      1)+4.DO*PI*(RT(I)+RO)**2*PHTW(I)+(RO+RT(I))**2*PHTW(I)**2+XTW(I)*(2HOFN2300
      2.DO*DXODW+XTW(I))                                                         HOFN2310
       HTWW=4.DO*PI**2*(RT(I)*RTW(I)+RT(I)*DRODW+RTW(I)*RO)+RTW(I)*RTWW(IHOFN2320
      1)+D2RDWW*RTW(I)+4.DO*PI*(RO+RT(I))*(DRODW+RTW(I))*PHTW(I)+DRODW*RTHOFN2330
      2WW(I)+2.DO*PI*(RT(I)+RO)**2*PHTWW(I)+(RO+RT(I))*(DRODW+RTW(I))*PHTHOFN2340
      3W(I)**2+(RO+RT(I))**2*PHTW(I)*PHTWW(I)+DXODW*XTWW(I)+XTW(I)*(D2XODHOFN2350
      4W+XTWW(I))                                                                HOFN2360
       HTWX=4.DO*PI**2*(RT(I)*(RTX(I)+DRODX)+RO*RTX(I))+RTW(I)*RTWX(I)+D2HOFN2370
      1RDXW*RTW(I)+DRODW*RTWX(I)+4.DO*PI*(RO+RT(I))*(DRODW+RTW(I))*PHTW(IHOFN2380
      2)+2.DO*PI*(RT(I)+RO)**2*PHTW(I)+(RO+RT(I))*(DRODX+RT(I))*PHTW(I)**HOFN2390
      32+(RO+RT(I))**2*PHTW(I)*PHTWX(I)+DXODW*XTWX(I)+XTW(I)*(D2XDWX+XTWXHOFN2400
      4(I))                                                                      HOFN2410
       GO TO 240                                                                 HOFN2420
230    CONTINUE                                                                  HOFN2430
       HTW=0.DO                                                                  HOFN2440
       HTWW=0.DO                                                                 HOFN2450
       HTWX=0.DO                                                                 HOFN2460
240    CONTINUE                                                                  HOFN2470
       IF (IINC.NE.5000) GO TO 250                                               HOFN2480
       WRITE (6,340) HTX,HTW,HTXX,HTXW,HTWX,HTWW                                 HOFN2490
       IINC=0                                                                    HOFN2500
250    CONTINUE                                                                  HOFN2510
       IINC=IINC+1                                                               HOFN2520
       IF (K.NE.1) GO TO 260                                                     HOFN2530
       HTXWW=RTWWX(I)*DRODX+RTX(I)*D3RWWX+RTX(I)*RTWWX(I)+XTX(I)*(XTXWW(IHOFN2540
      1)+D3XXWW)+DXODX*XTXWW(I)+(RO+RT(I))*PHTWX(I)                             HOFN2550
260    CONTINUE                                                                  HOFN2560
       HX(I)=DSQRT(HXO**2+HTX)                                                   HOFN2570
       HW(I)=DSQRT(HWO**2+HTW)                                                   HOFN2580
       HXXOLD(I)=HXX(I)                                                          HOFN2590
       HXX(I)=(HXO*DHXODX+HTXX)/HX(I)                                            HOFN2600
       HXW(I)=(HXO*DHXODW+HTXW)/HX(I)                                            HOFN2610
       IF (L.EQ.1) GO TO 270                                                     HOFN2620
       HWX(I)=(HWO*DHWODX+HTWX)/HW(I)                                            HOFN2630
       HWW(I)=(HWO*DHWODW+HTWW)/HW(I)                                            HOFN2640
       GO TO 280                                                                 HOFN2650
270    CONTINUE                                                                  HOFN2660
       HWX(I)=HWX(1)                                                             HOFN2670
       HWW(I)=0.DO                                                               HOFN2680
280    CONTINUE                                                                  HOFN2690
       IF (K.NE.1) GO TO 290                                                     HOFN2700
       HXWW(I)=(HXO*DHXDWW+HTXWW)/HX(I)                                          HOFN2710
290    CONTINUE                                                                  HOFN2720
300    CONTINUE                                                                  HOFN2730
       CALL DERIV (HX,ETA,IE,1,HXN)                                              HOFN2740
      CALL DERIV (HXN,ETA,IE,1,HXNN)
      CALL DERIV (HW,ETA,IE,1,HWN)
      CALL DERIV (HWN,ETA,IE,1,HWNN)
      RETURN
  310 CONTINUE
C
C     CALCULATE METRIC COEFFICIENTS FOR PROFILE POINTS WHEN TVC = NO
C
      DO 320 I=1,IE
      HX(I)=HX0
      HW(I)=HW0
      HXX(I)=DHX0DX
      HXW(I)=DHX0DW
      HWX(I)=DHW0DX
      HWW(I)=DHW0DW
      HXWW(I)=DHXDWW
      HXN(I)=0.D0
      HXNN(I)=0.D0
      HWN(I)=0.D0
      HWNN(I)=0.D0
  320 CONTINUE
      RETURN
C
  330 FORMAT (1H0,5HTX = ,E13.6,6H TW = ,E13.6,7H TXX = ,E13.6,6HTXW = ,
     1E13.6,7H TWX = ,E13.6,7H TWW = ,E13.6)
  340 FORMAT (1H0,4HR = ,E13.6,6H RX = ,E13.6,6H RW = ,E13.6,7H RXX = ,E
     113.6,7H RWX = ,E13.6,7H RWW = ,E13.6)
      END


      SUBROUTINE INIT
      IMPLICIT REAL*8 (A-H,O-Z)
      REAL*8 NOSE
      COMMON /ASSVAR/ IFL,KBL
      COMMON /CONVRG/ CONV,NIT1,NIT2,NIT3,NIT
      COMMON /DEPVAR/ F(2,101,3),FN(2,101,3),G(2,101,3),GN(2,101,3),T(2,
     1101,3),TN(2,101,3),Y(101),YOL(101)
      COMMON /FRSTRM/ RHOINF,PINF,TFS,UFS,AMUINF,CP,REINF,PR
      COMMON /GASPRP/ PRANDT(101)
      COMMON /GEOM/ ALPHA,NOSE,RNOSE,WLST,X,XX,WX
      COMMON /INJECT/ INJCT,NOINJ,MASTRN
      COMMON /INTEGR/ IE,IM,KEND,KEND2,KLX,K,L,NBLNT1,IND,KPRT,LPRT,KPR,
     1LPR
      COMMON /SOLPNT/ VW(101),GW(101),TW(101),GWN(101),FWN(101),FW(101),
     1TWN(101),XIW,XW,RW
      COMMON /SURFAS/ CWALL,CWIND,PEWIND,VWALL,TWALL,XTW(500),TWX(500),X
     1CI(500),CIX(500),TCONW,KCI,KTW
      COMMON /TRANSN/ KTRANS,KONSET,XIF,CHI2(101),CHIMAX,XBAR
```

DATE 05/12/77

```
      COMMON /TRBLNT/ ASTAR,AKSTAR,ALAMDA,YSUBL,EVSCTY(101),EDYLAW,EPLUSINIT 190
     1(101),LAMTRB                                                      INIT 200
      COMMON /TMPRTR/ TEMP(101),TP(101),RTW,TB                          INIT 210
      COMMON /WSOLVE/ DW,W                                              INIT 220
      COMMON /XICORD/ XI,XXI,DXI,XIOLD                                  INIT 230
      COMMON /XSOLVE/ XSTA(100),DXMAX,DX,DXOLD,DX1,NSOLVE               INIT 240
      COMMON /ZCOORD/ ETAINF,ETAFAC,ETA(101),DETA(101),ADTEST,KADETA    INIT 250
      COMMON /OLD/ VOLD(61),CVOLD(61),ROLD(61)                          INIT 260
      DATA BLUNT,SHARP/5HBLUNT,5HSHARP/                                 INIT 270
      READ (10) ALPHA,KLX                                               INIT 280
      ALPHA=ALPHA*DARCOS(-1.0D0)/180.0D0                                INIT 290
      WLST=0.5D0                                                        INIT 300
      ZWOLD=1.0D0                                                       INIT 310
      SINLST=0.0D0                                                      INIT 320
      X=0.0D0                                                           INIT 330
      DX=DX1                                                            INIT 340
      XX=0.0D0                                                          INIT 350
      XI=0.0D0                                                          INIT 360
      DXI=0.0D0                                                         INIT 370
      XXI=0.0D0                                                         INIT 380
      XIW=0.0D0                                                         INIT 390
      RW=0.0D0                                                          INIT 400
      XW=0.0D0                                                          INIT 410
      XIOLD=0.0D0                                                       INIT 420
      DXOLD=DX                                                          INIT 430
      XIF=0.0D0                                                         INIT 440
      IF (LAMTRB.EQ.2) XIF=1.0D0                                        INIT 450
      CWIND=CWALL                                                       INIT 460
      VWALL=0.0D0                                                       INIT 470
      DO 10 N=1,IE                                                      INIT 480
      PRANDT(N)=0.9D0                                                   INIT 490
      EPLUS(N)=0.0D0                                                    INIT 500
      CHI2(N)=0.0D0                                                     INIT 510
   10 EVSCTY(N)=0.0D0                                                   INIT 520
      IND=1                                                             INIT 530
      KBL=1                                                             INIT 540
      L=0                                                               INIT 550
      NIT=0                                                             INIT 560
      LPR=LPRT                                                          INIT 570
      KEND=KEND2                                                        INIT 580
      IF (ALPHA.LT.1.D-08) KEND=1                                       INIT 590
      IM=IE-1                                                           INIT 600
      DW=1.0D0                                                          INIT 610
      DETA(1)=0.0D0                                                     INIT 620
      ETA(1)=0.0D0                                                      INIT 630
      ETA(IE)=ETAINF                                                    INIT 640
      IF (ETAFAC.EQ.1.0D0) GO TO 20                                     INIT 650
      DETA(2)=ETAINF*(ETAFAC-1.0D0)/(ETAFAC**IM-1.0D0)                  INIT 660
      GO TO 30                                                          INIT 670
   20 DETA(2)=ETAINF/DFLOAT(IM)                                         INIT 680
   30 ETA(2)=DETA(2)                                                    INIT 690
```

DATE 05/12/77

```
      DO 40 I=3,IM
      DETA(I)=ETAFAC*DETA(I-1)
      ETA(I)=ETA(I-1)+DETA(I)
   40 CONTINUE
      DETA(IE)=DETA(IM)*ETAFAC
C
C     CALCULATE INITIAL PROFILES
C
      DO 70 I=1,2
      DO 60 N=1,3
      DO 50 J=1,IE
      F(I,J,N)=1.0D0-DEXP(-ETA(J))
      G(I,J,N)=.075D0*F(I,J,N)
      T(I,J,N)=RTW+(1.0D0-RTW)*F(I,J,N)
   50 CONTINUE
   60 CONTINUE
   70 CONTINUE
      DO 80 I=1,61
      VOLD(I)=0.D0
      CVOLD(I)=0.D0
      ROLD(I)=0.D0
   80 CONTINUE
      DO 90 J=1,IE
      Y(J)=0.0D0
      YOL(J)=0.0D0
      GW(J)=G(2,J,1)
      FW(J)=F(1,J,2)
   90 CONTINUE
      DO 110 I=1,2
      DO 100 N=1,3
      CALL DERIV3 (F,I,N,ETA,IE,1,FN)
      CALL DERIV3 (G,I,N,ETA,IE,1,GN)
      CALL DERIV3 (T,I,N,ETA,IE,1,TN)
  100 CONTINUE
  110 CONTINUE
      RETURN
      END


      SUBROUTINE INPUT
      IMPLICIT REAL*8 (A-H,O-Z)
      REAL*8 NOSE
      COMMON /CONVRG/ CONV,NIT1,NIT2,NIT3,NIT
      COMMON /EDGE/ UEDG,TEDG,VEDG,DUEGDX,DUEGDW,D2UEDW,DVEGDX,DVEGDW,RE
     1DG,DRDGDX,ZBOD,XEDG
      COMMON /FINDIF/ A(101),BB(101),B(101),CC(101),DD(101),D(101),E(101
     1),CRI
      COMMON /FRSTRM/ RHOINF,PINF,TFS,UFS,AMUINF,CP,REINF,PR
```

DATE 05/12/77

```
      COMMON /GEOM/ ALPHA,NOSE,RNOSE,WLST,X,XX,WX
      COMMON /INJECT/ INJCT,NOINJ,MASTRN
      COMMON /INTEGR/ IE,IM,KEND,KEND2,KLX,K,L,NBLNT1,IND,KPRT,LPRT,KPR,
     1LPR
      COMMON /STAG/ PSTAG,TSTAG,PNC,QWSTAG
      COMMON /SURFAS/ CWALL,CWIND,PEWIND,VWALL,TWALL,XTW(500),TWX(500),X
     1CI(500),CIX(500),TCONW,KCI,KTW
      COMMON /TITLE/ LABEL(20)
      COMMON /TMPRTR/ TEMP(101),TP(101),RTW,TB
      COMMON /TRANSN/ KTRANS,KONSET,XIF,CHI2(101),CHIMAX,XBAR
      COMMON /TRBLNT/ ASTAR,AKSTAR,ALAMDA,YSUBL,EVSCTY(101),EDYLAW,EPLUS
     1(101),LAMTRB
      COMMON /XSOLVE/ XSTA(100),DXMAX,DX,DXOLD,DX1,NSOLVE
      COMMON /ZCOORD/ ETAINF,ETAFAC,ETA(101),DETA(101),ADTEST,KADETA
      COMMON /TVCURV/ SFC
      DATA YES,NO/3HYES,3H NO/
      DATA BLUNT,SHARP/5HBLUNT,5HSHARP/
C
C     THE INPUT QUANTITIES ARE READ IN
C
      J=5
      READ (J,70) LABEL
      READ (J,50) IE
      READ (J,50) INJCT
      READ (J,50) KADETA
      READ (J,50) KEND2
      READ (J,50) KONSET
      READ (J,50) KPRT
      READ (J,50) KTRANS
      READ (J,50) LAMTRB
      READ (J,50) LPRT
      READ (J,50) NIT1
      READ (J,50) NIT2
      READ (J,50) NIT3
      READ (J,50) NOINJ
      READ (J,100) SFC
      READ (J,50) NSOLVE
      READ (J,60) ADTEST
      READ (J,60) AKSTAR
      READ (J,60) ALAMDA
      READ (J,60) ASTAR
      READ (J,60) CWALL
      READ (J,60) CONV
      READ (J,60) DXMAX
      READ (J,60) DX1
      READ (J,80) EDYLAW
      READ (J,60) ETAFAC
      READ (J,60) ETAINF
      READ (J,60) PR
      READ (J,60) RTW
      READ (J,60) TFS
      READ (J,60) CP
      READ (J,60) AMUINF
      READ (J,60) PSTAG
      READ (J,60) PINF
      READ (J,60) XBAR
      READ (J,60) UFS
      DO 10 I=1,NSOLVE
      READ (J,40) XSTA(I)
10    CONTINUE
      KTW=0
      KCI=0
      I=0
20    I=I+1
      READ (J,90,END=30) XTW(I),TWX(I),XCI(I),CIX(I)
      IF (XTW(I).EQ.0.0D0) XTW(I)=XCI(I)
      IF (XCI(I).EQ.0.0D0) XCI(I)=XTW(I)
      IF (CIX(I).EQ.0.0D0.AND.KCI.EQ.0) KCI=I-1
      GO TO 20
30    KTW=I-1
      IF (KCI.EQ.0) KCI=I-1
      MASTRN=0
      NOSE=BLUNT
      RNOSE=1.D0
      CRI=1.D0
      IF (KONSET.EQ.0) KONSET=NSOLVE
      IF (LAMTRB.EQ.2) KONSET=NSOLVE
      IF (INJCT.EQ.0) INJCT=NSOLVE
      IF (NOSE.EQ.SHARP.AND.INJCT.EQ.1) INJCT=2
      IF (INJCT.EQ.1) MASTRN=1
      IF (MASTRN.EQ.1) INJCT=NSOLVE
      IF (NOINJ.EQ.0) NOINJ=NSOLVE
C
C
      RETURN
C
C
C
40    FORMAT (F12.6)
50    FORMAT (49X,I3)
60    FORMAT (49X,E14.6)
70    FORMAT (20A4)
80    FORMAT (49X,A3)
90    FORMAT (4E12.6)
100   FORMAT (49X,A3)
      END


      SUBROUTINE INTER3 (X,X1,X2,X3,F1,F2,F3,F)
```

DATE 05/12/77

```
      IMPLICIT REAL*8(A-H,O-Z)                                          INT3  20
C                                                                       INT3  30
C     SUBROUTINE INTER3 INTERPOLATES FOR THE VALUE F CORRESPONDING TO   INT3  40
C     POINT X USING 3 POINT LAGRANGIAN INTERPOLATION.                   INT3  50
C                                                                       INT3  60
C     ASSUMES X1 .LE. X .LE. X3.                                        INT3  70
C                                                                       INT3  80
      A1=(X-X2)*(X-X3)                                                  INT3  90
      A2=(X-X1)*(X-X3)                                                  INT3 100
      A3=(X-X1)*(X-X2)                                                  INT3 110
      D1=(X1-X2)*(X1-X3)                                                INT3 120
      D2=(X2-X1)*(X2-X3)                                                INT3 130
      D3=(X3-X1)*(X3-X2)                                                INT3 140
      C1=A1/D1                                                          INT3 150
      C2=A2/D2                                                          INT3 160
      C3=A3/D3                                                          INT3 170
      F=C1*F1+C2*F2+C3*F3                                               INT3 180
      RETURN                                                            INT3 190
      END                                                               INT3 200


      SUBROUTINE INTER5 (X,X1,X2,X3,X4,X5,F1,F2,F3,F4,F5,F)             INT5  10
      IMPLICIT REAL*8 (A-H,O-Z)                                         INT5  20
C                                                                       INT5  30
C     SUBROUTINE INTER5 INTERPOLATES FOR THE VALUE F CORRESPONDING TO   INT5  40
C     POINT X USING 5 POINT LAGRANGIAN INTERPOLATION FORMULA.           INT5  50
C                                                                       INT5  60
C     ASSUMES X1 .LE. X .LE. X5.                                        INT5  70
C                                                                       INT5  80
      A1=(X-X2)*(X-X3)*(X-X4)*(X-X5)                                    INT5  90
      A2=(X-X1)*(X-X3)*(X-X4)*(X-X5)                                    INT5 100
      A3=(X-X1)*(X-X2)*(X-X4)*(X-X5)                                    INT5 110
      A4=(X-X1)*(X-X2)*(X-X3)*(X-X5)                                    INT5 120
      A5=(X-X1)*(X-X2)*(X-X3)*(X-X4)                                    INT5 130
      D1=(X1-X2)*(X1-X3)*(X1-X4)*(X1-X5)                                INT5 140
      D2=(X2-X1)*(X2-X3)*(X2-X4)*(X2-X5)                                INT5 150
      D3=(X3-X1)*(X3-X2)*(X3-X4)*(X3-X5)                                INT5 160
      D4=(X4-X1)*(X4-X2)*(X4-X3)*(X4-X5)                                INT5 170
      D5=(X5-X1)*(X5-X2)*(X5-X3)*(X5-X4)                                INT5 180
      C1=A1/D1                                                          INT5 190
      C2=A2/D2                                                          INT5 200
      C3=A3/D3                                                          INT5 210
      C4=A4/D4                                                          INT5 220
      C5=A5/D5                                                          INT5 230
      F=C1*F1+C2*F2+C3*F3+C4*F4+C5*F5                                   INT5 240
      RETURN                                                            INT5 250
      END                                                               INT5 260
```

DATE 05/12/77

```
      SUBROUTINE OUT1
      IMPLICIT REAL*8 (A-H,O-Z)
      REAL*8 NOSE
      COMMON /CONVRG/ CONV,NIT1,NIT2,NIT3,NIT
      COMMON /FINDIF/ A(101),BB(101),B(101),CC(101),DD(101),D(101),E(101
     1),CRI
      COMMON /FRSTRM/ RHOINF,PINF,TFS,UFS,AMUINF,CP,REINF,PR
      COMMON /GEOM/ ALPHA,NOSE,RNOSE,WLST,X,XX,WX
      COMMON /INJECT/ INJCT,NOINJ,MASTRN
      COMMON /INTEGR/ IE,IM,KEND,KEND2,KLX,K,L,NBLNT1,IND,KPRT,LPRT,KPR,
     1LPR
      COMMON /REF/ PREF,TREF,AMUREF
      COMMON /STAG/ PSTAG,TSTAG,PNC,QWSTAG
      COMMON /SURFAS/ CWALL,CWIND,PEWIND,VWALL,TWALL,XTW(500),TWX(500),X
     1CI(500),CIX(500),TCONW,KCI,KTW
      COMMON /TMPRTR/ TEMP(101),TP(101),RTW,TB
      COMMON /TRBLNT/ ASTAR,AKSTAR,ALAMDA,YSUBL,EVSCTY(101),EDYLAW,EPLUS
     1(101),LAMTRB
      COMMON /XICORD/ XI,XXI,DXI,XIOLD
      COMMON /XSOLVE/ XSTA(100),DXMAX,DX,DXOLD,DX1,NSOLVE
      COMMON /ZCOORD/ ETAINF,ETAFAC,ETA(101),DETA(101),ADTEST,KADETA
      COMMON /TITLE/ LABEL(20)
      DATA BLUNT,SHARP/5HBLUNT,5HSHARP/
      ATTAK=ALPHA*180.0D0/DARCOS(-1.0D0)
      WRITE (6,50)
      WRITE (6,140) (LABEL(I),I=1,20)
      WRITE (6,40)
      WRITE (6,60) PSTAG
      WRITE (6,70) TSTAG
      WRITE (6,80) PINF
      WRITE (6,100) UFS
      WRITE (6,90) TFS
      WRITE (6,150) RHOINF
      WRITE (6,160) AMUINF
      WRITE (6,110) RTW
      WRITE (6,120) ATTAK
      WRITE (6,170) CP
      WRITE (6,180) REINF
      WRITE (6,190) PR
      WRITE (6,20)
      DO 10 I=1,NSOLVE
      WRITE (6,30) I,XSTA(I)
   10 CONTINUE
      WRITE (6,130)
C
C
```

DATE 05/12/77

```
      RETURN
C
C
C
20    FORMAT (26X,45HPOINTS AT WHICH A SOLUTION IS TO BE OBTAINED://28X,
     11HI,6X,7HXSTA(I)/)
30    FORMAT (26X,I3,5X,F9.6)
40    FORMAT (//26X,42HFREE STREAM, STAGNATION, AND VEHICLE DATA://)
50    FORMAT (46X,40HTHREE-DIMENSIONAL BOUNDARY LAYER PROGRAM/65X,3HFOR/
     154X,25HLAMINAR OR TURBULENT FLOW/64X,4HWITH/53X,25HSURFACE CURVATU
     2RE EFFECTS/60X,12HDEVELOPED BY/50X,32HAEROSPACE ENGINEERING DEPART
     3MENT/40X,51HVIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY/56
     4X,21HBLACKSBURG, VA. 24061///)
60    FORMAT (26X,7HPSTAG =,E13.6,5H PSIA)
70    FORMAT (26X,7HTSTAG =,E13.6,6H DEG.R)
80    FORMAT (26X,7HPINF  =,E13.6,5H PSIA)
90    FORMAT (26X,7HTINF  =,E13.6,6H DEG.R)
100   FORMAT (26X,7HUINF  =,E13.6,7H FT/SEC)
110   FORMAT (26X,7HTW/TO =,E13.6)
120   FORMAT (26X,7HALPHA =,E13.6,5H DEG.)
130   FORMAT (1HO)
140   FORMAT (26X,20A4)
150   FORMAT (26X,7HRHOINF=,E13.6,12H SLUGS/FT**3)
160   FORMAT (26X,7HAMUINF=,E13.6,13H SLUGS/FT-SEC)
170   FORMAT (26X,7HCP    =,E13.6,21H FT**2/(SEC**2-DEG.R))
180   FORMAT (26X,7HREINF =,E13.6)
190   FORMAT (26X,7HPR    =,E13.6)
      END


      SUBROUTINE OUT2
      IMPLICIT REAL*8 (A-H,O-Z)
      REAL*8 NOSE
      COMMON /CONVRG/ CONV,NIT1,NIT2,NIT3,NIT
      COMMON /DEPVAR/ F(2,101,3),FN(2,101,3),G(2,101,3),GN(2,101,3),T(2,
     1101,3),TN(2,101,3),Y(101),YOL(0101)
      COMMON /EDGE/ UEDG,TEDG,VEDG,DUEGDX,DUEGDW,D2UEGW,DVEGDX,DVEGDW,RE
     1DG,DRDGDX,ZBOD,XEDG
      COMMON /EDG2/ TE2,UE2,VE2,DUE2DX,DVE2DX,DUE2DW,D2UE2W,DVE2DW,RE2,X
     12
      COMMON /GASPRP/ PRANDT(101)
      COMMON /GEOM/ ALPHA,NOSE,RNOSE,WLST,X,XX,WX
      COMMON /INJECT/ INJCT,NOINJ,MASTRN
      COMMON /INTEGR/ IE,IM,KEND,KEND2,KLX,K,L,NBLNT1,IND,KPRT,LPRT,KPR,
     1LPR
      COMMON /OUTPUT/ CFWEDG,CFWINF,CFXEDG,CFXINF,DEL,QW,QWINF,QWOQWO,S,
     1TAUETA,TAUX,DELSTX,DELPHI,THETAX,THEPHI,REX2
      COMMON /SOLPNT/ VW(101),GW(101),TW(101),GWN(101),FWN(101),FW(101),
     1TWN(101),XIW,XW,RW
      COMMON /SURFAS/ CWALL,CWIND,PEWIND,VWALL,TWALL,XTW(500),TWX(500),X
     1CI(500),CIX(500),TCONW,KCI,KTW
      COMMON /TMPRTR/ TEMP(101),TP(101),RTW,TB
      COMMON /TRANSN/ KTRANS,KONSET,XIF,CHI2(101),CHIMAX,XBAR
      COMMON /TRBLNT/ ASTAR,AKSTAR,ALAMDA,YSUBL,EVSCTY(101),EDYLAW,EPLUS
     1(101),LAMTRB
      COMMON /XICORD/ XI,XXI,DXI,XIOLD
      COMMON /XSOLVE/ XSTA(100),DXMAX,DX,DXOLD,DX1,NSOLVE
      COMMON /ZCOORD/ ETAINF,ETAFAC,ETA(101),DETA(101),ADTEST,KADETA
      COMMON /OUTSP/ DUDX,DVDX,DRDX
      COMMON /HETA/ HX(101),HW(101),HXX(101),HXW(101),HXWW(101),HXN(101)
     1,HXNN(101),HWX(101),HWW(101),HWN(101),HWNN(101)
      DATA BLUNT,SHARP/5HBLUNT,5HSHARP/
C
C
      IF (L.NE.1.AND.L.NE.LPR) GO TO 40
C
      WRITE (6,50) X2,RE2,WX
      WRITE (6,60) XI,DXI,CWALL,NIT
      WRITE (6,200) HX(1),HW(1)
      WRITE (6,70)
      WRITE (6,80) TE2,UE2,VE2,DUDX,DVDX,DUE2DW,DVE2DW
      WRITE (6,180) REX2
      IF (L.EQ.1.AND.NOSE.EQ.SHARP) GO TO 10
      WRITE (6,190)
      WRITE (6,90)
      WRITE (6,100) CFXINF,CFXEDG,CFWINF,CFWEDG,QWINF,CHIMAX
      WRITE (6,110)
      WRITE (6,120) TAUX,DELSTX,THETAX,TAUETA,DELPHI,THEPHI,QW,DEL
      IF (X.GE.XSTA(KONSET).AND.KONSET.NE.NSOLVE) WRITE (6,170) XIF
      WRITE (6,160)
10    CONTINUE
      IF (K.NE.1.AND.K.NE.KPR) GO TO 30
      IF (K.NE.1.OR.KPRT.EQ.1) KPR=KPR+KPRT
      WRITE (6,140)
      DO 20 N=1,IE,2
      WRITE (6,150) ETA(N),Y(N),F(2,N,2),FN(2,N,2),G(2,N,2),GN(2,N,2),T(
     12,N,2),TN(2,N,2),VW(N),EPLUS(N)
20    CONTINUE
30    CONTINUE
      WRITE (6,130)
40    CONTINUE
C
C
      RETURN
C
C
C
50    FORMAT (10X,6HX0    =,E13.6,4X,6HR     =,E13.6,4X,6HPHI   =,F6.2,5H D
     1EG.)
```

```
60      FORMAT (10X,6HXI    =,E13.6,4X,6HDXI    =,E13.6,4X,6HCWALL=,E13.6,5X,OUT2 700
       16HNIT   =,I3)                                                      OUT2 710
70      FORMAT (10X,30HNONDIMENSIONAL EDGE PROPERTIES/)                    OUT2 720
80      FORMAT (10X,6HTE    =,E13.6,5X,6HUE    =,E13.6,5X,6HVE    =,E13.6/10XOUT2 730
       1,6HDUEDX=,E13.6,5X,6HDVEDX=,E13.6,5X,6HDUEDW=,E13.6,5X,6HDVEDW=,E1OUT2 740
       23.6/)                                                              OUT2 750
90      FORMAT (10X,40HNONDIMENSIONAL BOUNDARY LAYER PARAMETERS/)          OUT2 760
100     FORMAT (10X,7HCFXINF=,E13.6,5X,7HCFXEDG=,E13.6,5X,7HCFWINF=,E13.6,OUT2 770
       15X,7HCFWEDG=,E13.6/10X,7HQW     =,E13.6,5X,7HCHIMAX=,E13.6//)      OUT2 780
110     FORMAT (10X,37HDIMENSIONAL BOUNDARY LAYER PARAMETERS/)             OUT2 790
120     FORMAT (10X,27HLONGITUDINAL SKIN FRICTION=,E13.6,4H PSF,5X,12HDELTOUT2 800
       1A*(X)  =,E13.6,5X,11HTHETA(X)  =,E13.6/10X,27HTRANSVERSE SKIN FRICOUT2 810
       2TION  =,E13.6,4H PSF,5X,12HDELTA*(PHI)=,E13.6,5X,11HTHETA(PHI)=,E1OUT2 820
       33.6/10X,27HWALL HEAT TRANSFER RATE    =,E13.6,4H BTU,5X,12HDELTA (FOUT2 830
       4T) =,E13.6)                                                        OUT2 840
130     FORMAT (1H0,48X,3(5H*****,10X)//)                                  OUT2 850
140     FORMAT (7X,3HETA,11X,1HY,10X,1HF,10X,2HFN,9X,1HG,10X,2HGN,9X,1HT,1OUT2 860
       10X,2HTN,9X,1HV,8X,5HEPLUS/)                                        OUT2 870
150     FORMAT (1X,F12.6,3X,E9.3,1X,6(F10.6,1X),E10.3,1X,E10.3)            OUT2 880
160     FORMAT (1H0)                                                       OUT2 890
170     FORMAT (1H0,9X,34HTRANSITION INTERMITTENCY FACTOR =  ,E12.6)       OUT2 900
180     FORMAT (10X,28HLOCAL EDGE REYNOLDS NUMBER =,E12.6/)                OUT2 910
190     FORMAT (1H )                                                       OUT2 920
200     FORMAT (10X,6HHX    =,E13.6,4X,6HHW    =,E13.6//)                  OUT2 930
        END                                                                OUT2 940


        SUBROUTINE PHIMOM (LC)                                             PHIM  10
        IMPLICIT REAL*8(A-H,O-Z)                                           PHIM  20
        REAL*8 NOSE                                                        PHIM  30
        COMMON /DEPVAR/ F(2,101,3),FN(2,101,3),G(2,101,3),GN(2,101,3),T(2,PHIM  40
       1101,3),TN(2,101,3),Y(101),YOL(101)                                 PHIM  50
        COMMON /GEOM/ ALPHA,NOSE,RNOSE,WLST,X,XX,WX                        PHIM  60
        COMMON /IECOEF/ B1,B2,B3,B4,B5,B6,AL1,AL2,AL3,XLAMB                PHIM  70
        COMMON /INTEGR/ IE,IM,KEND,KEND2,KLX,K,L,NBLNT1,IND,KPRT,LPRT,KPR,PHIM  80
       1LPR                                                                PHIM  90
        COMMON /PDECOF/ A0(101),A1(101),A2(101),A3(101),A4(101),A5(101)    PHIM 100
        COMMON /SOLPNT/ VW(101),GW(101),TW(101),GWN(101),FWN(101),FW(101),PHIM 110
       1TWN(101),XIW,XW,RW                                                 PHIM 120
        COMMON /TRANSN/ KTRANS,KONSET,XIF,CHI2(101),CHIMAX,XBAR            PHIM 130
        COMMON /TRBLNT/ ASTAR,AKSTAR,ALAMDA,YSUBL,EVSCTY(101),EDYLAW,EPLUSPHIM 140
       1(101),LAMTRB                                                       PHIM 150
        COMMON /XICORD/ XI,XXI,DXI,XIOLD                                   PHIM 160
        COMMON /ZCOORD/ ETAINF,ETAFAC,ETA(101),DETA(101),ADTEST,KADETA     PHIM 170
        COMMON /NCOORD/ XN(101),ETAX,ETAW                                  PHIM 180
        COMMON /EDGW/ UEW,VEW,TEW,DUEWDX,DUEWDW,D2UEWW,DVEWDX,DVEWDW,REW   PHIM 190
        COMMON /HETA/ HX(101),HW(101),HXX(101),HXW(101),HXWW(101),HXN(101)PHIM 200
       1,HXNN(101),HWX(101),HWW(101),HWN(101),HWNN(101)                    PHIM 210
```

```
      COMMON /GECOEF/ HXOHXO(101),HWOHWO(101),BHV(101),BHM(101),BHT(101)PHIM 220
     1,XKX(101),XKW(101),BMX(101),BMW(101),XLX(101),XLW(101),BAFF(101),BPHIM 230
     2AGF(101),BAFG(101),BAGG(101),XKXOAL(101)                         PHIM 240
      COMMON /STGEDG/ DVEGXW,DVEWXW,DVE2XW,DVEGXX,DVEWXX,DVE2XX,DVGXXW,DPHIM 250
     1VWXXW,DV2XXW,DHWOXW,DHXOXW                                       PHIM 260
      COMMON /INTGRL/ XIFNX(101),XIFNW(101),XIFNXX(101),XIFNWW(101)     PHIM 270
      COMMON /PHI/ DAL3DX,DAL3DW,ALB6                                   PHIM 280
      DIMENSION ROMUI(101), ROMUIN(101)                                 PHIM 290
      DATA SHARP,BLUNT/5HSHARP,5HBLUNT/                                 PHIM 300
C                                                                       PHIM 310
C     SUBROUTINE PHIMOM SETS UP THE COEFFICIENTS OF THE PARTIAL         PHIM 320
C     DIFFERENTIAL PHI MOMENTUM EQUATION                                PHIM 330
C                                                                       PHIM 340
      DO 10 J=1,IE                                                      PHIM 350
      ROMUI(J)=1.0D0+XIF*EPLUS(J)                                       PHIM 360
10    CONTINUE                                                          PHIM 370
      CALL DERIV (ROMUI,ETA,IE,1,ROMUIN)                                PHIM 380
      CALL PRESSI (2)                                                   PHIM 390
      DO 40 J=1,IE                                                      PHIM 400
      A0(J)=ROMUI(J)                                                    PHIM 410
      A1(J)= ROMUIN(J)-BHM(J)*ROMUI(J)-VW(J)                            PHIM 420
      A2(J)=-BMW(J)*ROMUI(J)                                            PHIM 430
      A4(J)=-HX(1)*XI*FW(J)/2.D0/HX(J)**2                               PHIM 440
      A5(J)=0.D0                                                        PHIM 450
      IF (LC.EQ.1) GO TO 30                                             PHIM 460
      IF (LC.EQ.2) GO TO 20                                             PHIM 470
      A2S=-AL2**2*B6/HWOHWO(J)**2/2.D0+AL2*(XLW(J)-XLW(1))/2.D0+BAGG(J) PHIM 480
      A2(J)=A2(J)+2.D0*A2S*GW(J)+ROMUIN(J)*HWN(J)/HW(J)-.5D0*B5*FW(J)/HXPHIM 490
     1OHXO(J)**2-.5D0*XKW(1)*FW(J)/HXOHXO(J)**2                        PHIM 500
      A3(J)=B5/2.D0*AL3/HXOHXO(J)**2+AL1*AL3*B6/2.D0/HWOHWO(J)**2-XKX(J)PHIM 510
     1*HWOHWO(J)*(1.D0-FW(J)**2)/2.D0/HXOHXO(J)**2/AL2+XKW(1)/2.D0*AL3/HPHIM 520
     2XOHXO(J)**2-AL2*AL3**2*(XLW(J)-XLW(1))/2.D0-BAFG(J)*(1.D0-FW(J)**2PHIM 530
     3)-BAGG(J)*AL3**2-XI*XIFNW(J)/2.D0/HW(1)/UEW**2+.5D0*XI*DAL3DX/HXOHPHIM 540
     4XO(J)**2/HX(1)+AL1*XI*.5D0*DAL3DW/HW(1)/HWOHWO(J)**2-A2S*GW(J)**2 PHIM 550
      A5(J)=-HW(1)*XI*AL2*GW(J)/2.D0/HW(J)**2                           PHIM 560
      GO TO 40                                                          PHIM 570
20    CONTINUE                                                          PHIM 580
      A2(J)=A2(J)-XKW(1)*FW(J)/2.D0/HXOHXO(J)**2-B5*FW(J)/2.D0/HXOHXO(J)PHIM 590
     1**2+BAGG(J)*GW(J)+ROMUIN(J)*HWN(J)/HW(J)-ALB6/HWOHWO(J)**2*GW(J)/2PHIM 600
     2.D0                                                               PHIM 610
      A3(J)=B5/2.D0/HXOHXO(J)**2+XKW(1)/HXOHXO(J)**2/2.D0-XKXOAL(J)*(1.DPHIM 620
     10-FW(J)**2)*HWOHWO(J)/2.D0/HXOHXO(J)**2-BAFG(J)*(1.D0-FW(J)**2)-BAPHIM 630
     2GG(J)*AL3**2-XI*XIFNW(J)/HW(1)/UEW/DVEWDW+.5D0*ALB6/HWOHWO(J)**2  PHIM 640
      GO TO 40                                                          PHIM 650
30    CONTINUE                                                          PHIM 660
      A2S=-.5D0*AL1*B6                                                  PHIM 670
      A2(J)=-.5D0*FW(J)*B5/HXOHXO(J)**2-.5D0*FW(J)/HXOHXO(J)**2+2.D0*A2SPHIM 680
     1*GW(J)                                                            PHIM 690
      A3(J)=AL1*B6/2.D0-XKXOAL(J)*(1.D0-FW(J)**2)/2.D0/HXOHXO(J)**2+(BAGPHIM 700
     1F(J)+BAGG(J))/DUEWDX**1.5D0-3.D0*XIFNW(J)/DSQRT(2.D0*DUEWDX)-DSQRTPHIM 710
     2(2.D0*DUEWDX)*XIFNXX(J)/HWX(1)*DVEWDX-(DVEWXW*2.D0/HWX(1)/DUEWDX-APHIM 720
     3L1**2)*XIFNW(J)/DSQRT(2.D0*DUEWDX)-AL1*XIFNWW(J)/DSQRT(2.D0*DUEWDX
     4)*HWX(1)+.5D0*B5/HXOHXO(J)**2+.5D0/HXOHXO(J)**2-A2S*GW(J)**2
      A4(J)=0.D0
      A5(J)=-AL1*GW(J)/2.D0/HWX(1)
40    CONTINUE
      IRITE=0
      IF (IRITE.EQ.0) GO TO 60
      WRITE (6,70)
      DO 50 I=1,IE,5
      WRITE (6,80) A0(I),A1(I),A2(I),A3(I),A4(I),A5(I)
50    CONTINUE
60    CONTINUE
      RETURN
C
70    FORMAT (1H0,15HIN PHIMOM A0-A1)
80    FORMAT (1H0,6E13.6)
      END


      SUBROUTINE PRESSI (IDIREC)
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON /INTEGR/ IE,IM,KEND,KEND2,KLX,K,L,NBLNT1,IND,KPRT,LPRT,KPR,
     1LPR
      COMMON /INTGRL/ XIFNX(101),XIFNW(101),XIFNXX(101),XIFNWW(101)
      DO 10 I=1,IE
      XIFNX(I)=0.D0
      XIFNW(I)=0.D0
      XIFNXX(I)=0.D0
      XIFNWW(I)=0.D0
10    CONTINUE
      RETURN
      END


      SUBROUTINE PROPTY
      IMPLICIT REAL*8 (A-H,O-Z)
      REAL*8 NOSE
      COMMON /DEPVAR/ F(2,101,3),FN(2,101,3),G(2,101,3),GN(2,101,3),T(2,
     1101,3),TN(2,101,3),Y(101),YOL(101)
      COMMON /EDG2/ TE2,UE2,VE2,DUE2DX,DVE2DX,DUE2DW,D2UE2W,DVE2DW,RE2,X
     12
      COMMON /FRSTRM/ RHOINF,PINF,TFS,UFS,AMUINF,CP,REINF,PR
      COMMON /GASPRP/ PRANDT(101)
      COMMON /GEOM/ ALPHA,NOSE,RNOSE,WLST,X,XX,WX
      COMMON /INTEGR/ IE,IM,KEND,KEND2,KLX,K,L,NBLNT1,IND,KPRT,LPRT,KPR,
```

DATE 05/12/77

```
     1LPR                                                                PROP 120
      COMMON /OUTPUT/ CFWEDG,CFWINF,CFXEDG,CFXINF,DEL,QW,QWINF,QWOQWO,S,PROP 130
     1TAUETA,TAUX,DELSTX,DELPHI,THETAX,THEPHI,REX2                        PROP 140
      COMMON /STAG/ PSTAG,TSTAG,PNC,QWSTAG                               PROP 150
      COMMON /SURFAS/ CWALL,CWIND,PEWIND,VWALL,TWALL,XTW(500),TWX(500),XPROP 160
     1CI(500),CIX(500),TCONW,KCI,KTW                                      PROP 170
      COMMON /TMPRTR/ TEMP(101),TP(101),RTW,TB                           PROP 180
      COMMON /TRANSN/ KTRANS,KONSET,XIF,CHI2(101),CHIMAX,XBAR            PROP 190
      COMMON /TRBLNT/ ASTAR,AKSTAR,ALAMDA,YSUBL,EVSCTY(101),EDYLAW,EPLUSPROP 200
     1(101),LAMTRB                                                       PROP 210
      COMMON /XICORD/ XI,XXI,DXI,XIOLD                                   PROP 220
      COMMON /XSOLVE/ XSTA(100),DXMAX,DX,DXOLD,DX1,NSOLVE                PROP 230
      COMMON /ZCOORD/ ETAINF,ETAFAC,ETA(101),DETA(101),ADTEST,KADETA     PROP 240
      DIMENSION U(101), T2(101), Z2(101), F2(101), G2(101)              PROP 250
      DATA BLUNT,SHARP/5HBLUNT,5HSHARP/                                  PROP 260
      IF (NOSE.EQ.SHARP) S=X/XSTA(NSOLVE)                                PROP 270
      IF (NOSE.EQ.BLUNT) S=X/RNOSE                                       PROP 280
      DO 10 J=1,IE                                                       PROP 290
      F2(J)=F(2,J,2)                                                     PROP 300
      G2(J)=G(2,J,2)                                                     PROP 310
      IF (K.EQ.1) G2(J)=0.0D0                                            PROP 320
      T2(J)=T(2,J,2)                                                     PROP 330
   10 CONTINUE                                                           PROP 340
      LPR1=LPR                                                           PROP 350
      LPR=-10                                                            PROP 360
      LPR=LPR1                                                           PROP 370
C                                                                        PROP 380
C     THE OUTPUT QUANTITIES ARE CALCULATED                               PROP 390
C                                                                        PROP 400
      IF (L.EQ.1.AND.NOSE.EQ.SHARP) GO TO 50                             PROP 410
C                                                                        PROP 420
C     CALCULATE PHYSICAL NORMAL DISTANCE PROFILE                         PROP 430
C                                                                        PROP 440
      DELSTX=0.0D0                                                       PROP 450
      DELPHI=0.0D0                                                       PROP 460
      THETAX=0.0D0                                                       PROP 470
      THEPHI=0.0D0                                                       PROP 480
      CHIMAX=0.0D0                                                       PROP 490
      CHI2(1)=0.0D0                                                      PROP 500
      G2IE=G(2,IE,2)                                                     PROP 510
      IF (DABS(G2IE).LT.1.D-10) G2IE=1.0D0                               PROP 520
      Y(1)=0.0D0                                                         PROP 530
      YOL(1)=0.0D0                                                       PROP 540
      EPSVD=1.D0/DSQRT(REINF)                                            PROP 550
      IF (XI.EQ.0.0D0) YTRANS=1.0D0/PNC*EPSVD                            PROP 560
      IF (XI.GT.0.0D0) YTRANS=DSQRT(XI/(2.D0*UE2))*EPSVD                 PROP 570
C                                                                        PROP 580
C     CALCULATE DISPLACEMENT AND MOMENTUM THICKNESSES                    PROP 590
C                                                                        PROP 600
      DO 20 J=2,IE                                                       PROP 610
      Y(J)=Y(J-1)+YTRANS*DETA(J)                                         PROP 620
      YOL(J)=Y(J)/XSTA(NSOLVE)
      DELSTX=DELSTX+YTRANS*((1.0D0-F(2,J,2))+(1.0D0-F(2,J-1,2)))*DETA(J)
     1/2.D0
      THETAX=THETAX+YTRANS*(F(2,J,2)*(1.0D0-F(2,J,2))+F(2,J-1,2)*(1.0D0-
     1F(2,J-1,2)))*DETA(J)/2.0D0
      THEPHI=THEPHI+YTRANS*(G(2,J,2)/G2IE*(1.0D0-G(2,J,2)/G2IE)+G(2,J-1,
     12)/G2IE*(1.0D0-G(2,J-1,2)/G2IE))*DETA(J)/2.0D0
      DELPHI=DELPHI+YTRANS*((1.0D0-G(2,J,2)/G2IE)+(1.0D0-G(2,J,2)/G2IE))
     1*DETA(J)/2.D0
      CHI2(J)=Y(J)**2*FN(2,J,2)*UE2*(1.0D0/YTRANS)
      IF (CHI2(J).GT.CHIMAX) CHIMAX=CHI2(J)
   20 CONTINUE
      IF (G(2,IE,2).EQ.0.D0) DELPHI=0.D0
C
C     CALCULATE BOUNDARY LAYER THICKNESS
C
      DO 30 N=1,IE
      U(N)=DSQRT((F(2,N,2)**2+G(2,N,2)**2)/(F(2,IE,2)**2+G(2,IE,2)**2))
      IF (K.EQ.1) U(N)=F(2,N,2)
      IF (U(N).GE.0.995D0) NN=N-1
      IF (U(N).GE.0.995D0) GO TO 40
   30 CONTINUE
   40 DEL=Y(NN)+(Y(NN+1)-Y(NN))*(0.995D0-U(NN))/(U(NN+1)-U(NN))
C
   50 CONTINUE
      REX2=REINF*UE2*X
      IF (EPSVD.LT.1.D-25) EPSVD=0.D0
      CFXINF=0.D0
      CFWINF=0.D0
      CFXEDG=0.D0
      CFWEDG=0.D0
      QWINF=0.D0
      IF (X.EQ.0.0D0) GO TO 60
C
C     CALCULATE SURFACE PARAMETERS
C
      DWDETA=GN(2,1,2)
      IF (K.EQ.1.OR.K.EQ.KEND) DWDETA=0.0D0
      CFXINF=2.0D0*EPSVD*UE2**1.5*FN(2,1,2)*DSQRT(2.D0/XI)
      CFWINF=2.0D0*EPSVD*UE2**1.5*DWDETA*DSQRT(2.D0/XI)
      CFXEDG=2.0D0*EPSVD*FN(2,1,2)*DSQRT(2.D0/XI)
      CFWEDG=2.0D0*EPSVD*UE2**1.5*DWDETA*DSQRT(2.D0/XI)
      QWINF=EPSVD*TE2*UE2**.5*DSQRT(2.D0/XI)*TN(2,1,2)/PR
      QWINF=QWINF*(-1.0D0)
      TAUX=CFXINF*RHOINF*UFS**2/2.0D0
      TAUETA=CFWINF*RHOINF*UFS**2/2.0D0
      QW=QWINF*RHOINF*UFS**3*1.286D-03/PR
      IF (L.EQ.2.AND.NOSE.EQ.SHARP.AND.K.EQ.1) QWSTAG=QW
      QWOQWO=QW/QWSTAG
      GO TO 70
   60 IF (NOSE.EQ.SHARP) GO TO 70
```

```
C
C     CALCULATE HEAT TRANSFER FOR A BLUNT CONE STAGNATION POINT
C
      QWINF=EPSVD*TE2*DSQRT(2.D0*DUE2DX)*TN(2,1,2)/PR
      QWINF=QWINF*(-1.0D0)
      QW=QWINF*RHOINF*UFS**3*1.286D-03/PR
      QWSTAG=QW
      QWOQWO=1.0D0
C
70    CONTINUE
      RETURN
      END


      SUBROUTINE SOLVE (W,WN,EE1,FF1,EDGBC)
      IMPLICIT REAL*8(A-H,O-Z)
      REAL*8 NOSE
      COMMON /FINDIF/ A(101),BB(101),B(101),CC(101),DD(101),D(101),E(101
     1),CRI
      COMMON /INTEGR/ IE,IM,KEND,KEND2,KLX,KDUM,L,NBLNT1,IND,KPRT,LPRT,K
     1PR,LPR
      COMMON /ZCOORD/ ETAINF,ETAFAC,ETA(101),DETA(101),ADTEST,KADETA
      DIMENSION EE(101), FF(101), W(2,101,3), WN(2,101,3)
C
C     SUBROUTINE SOLVE CALCULATES THE SOLUTION OF A GENERAL PARABOLIC
C     PARTIAL DIFFERENTIAL EQUATION WHEN THE THE P.D.E. IS REPRESENTED
C     BY A SYSTEM OF IMPLICIT, THREE-POINT FINITE-DIFFERENCE EQUATIONS.
C     THE THOMAS ALGORITHM AS SOLVED BY RICHTMEYER IS USED TO SOLVE THE
C     SYSTEM.
C
      EE(1)=EE1
      FF(1)=FF1
      DO 10 J=2,IM
      AP=A(J)
      BP=B(J)
      CP=CC(J)
      DP=D(J)
      EE(J)=-CP/(BP+AP*EE(J-1))
      FF(J)=(DP-AP*FF(J-1))/(BP+AP*EE(J-1))
10    CONTINUE
      W(2,IE,2)=EDGBC
      K=IM
      DO 20 J=2,IE
      W(2,K,2)=EE(K)*W(2,K+1,2)+FF(K)
      K=K-1
20    CONTINUE
      CALL DERIV3 (W,2,2,ETA,IE,1,WN)
      RETURN
```

DATE 05/12/77

```
      END                                                               SOLV 350


      SUBROUTINE VCALC (LC)                                             VCAL  10
      IMPLICIT REAL*8(A-H,O-Z)                                          VCAL  20
      REAL*8 NOSE                                                       VCAL  30
      COMMON /DEPVAR/ F(2,101,3),FN(2,101,3),G(2,101,3),GN(2,101,3),T(2,VCAL  40
     1101,3),TN(2,101,3),Y(101),YOL(101)                                VCAL  50
      COMMON /HETA/ HX(101),HW(101),HXX(101),HXW(101),HXWW(101),HXN(101)VCAL  60
     1,HXNN(101),HWX(101),HWW(101),HWN(101),HWNN(101)                   VCAL  70
      COMMON /IECOEF/ B1,B2,B3,B4,B5,B6,AL1,AL2,AL3,XLAMB               VCAL  80
      COMMON /GECOEF/ HXOHXO(101),HWOHWO(101),BHV(101),BHM(101),BHT(101)VCAL  90
     1,XKX(101),XKW(101),BMX(101),BMW(101),XLX(101),XLW(101),BAFF(101),BVCAL 100
     2AGF(101),BAFG(101),BAGG(101),XKXOAL(101)                          VCAL 110
      COMMON /INTEGR/ IE,IM,KEND,KEND2,KLX,K,L,NBLNT1,IND,KPRT,LPRT,KPR,VCAL 120
     1LPR                                                               VCAL 130
      COMMON /SOLPNT/ VW(101),GW(101),TW(101),GWN(101),FWN(101),FW(101),VCAL 140
     1TWN(101),XIW,XW,RW                                                VCAL 150
      COMMON /SURFAS/ CWALL,CWIND,PEWIND,VWALL,TWALL,XTW(500),TWX(500),XVCAL 160
     1CI(500),CIX(500),TCONW,KCI,KTW                                    VCAL 170
      COMMON /WSOLVE/ DW,WCOORD                                         VCAL 180
      COMMON /XICORD/ XI,XXI,DXI,XIOLD                                  VCAL 190
      COMMON /XSOLVE/ XSTA(100),DXMAX,DX,DXOLD,DX1,NSOLVE               VCAL 200
      COMMON /ZCOORD/ ETAINF,ETAFAC,ETA(101),DETA(101),ADTEST,KADETA    VCAL 210
      COMMON /GEOM/ ALPHA,NOSE,RNOSE,WLST,X,XX,WX                       VCAL 220
      COMMON /STGEOG/ DVEGXW,DVEWXW,DVE2XW,DVEGXX,DVEWXX,DVE2XX,DVGXXW,DVCAL 230
     1VWXXW,DV2XXW,DHWOXW,DHXOXW                                        VCAL 240
      COMMON /EDGW/ UEW,VEW,TEW,DUEWDX,DUEWDW,D2UEWW,DVEWDX,DVEWDW,REW  VCAL 250
      COMMON /CONVRG/ CONV,NIT1,NIT2,NIT3,NIT                           VCAL 260
      DIMENSION V(101)                                                  VCAL 270
      DATA SHARP,BLUNT/5HSHARP,5HBLUNT/                                 VCAL 280
C                                                                       VCAL 290
C     THIS SUBROUTINE CALCULATES THE VALUE OF V                         VCAL 300
C                                                                       VCAL 310
      DO 10 J=1,IE                                                      VCAL 320
      V(J)=0.D0                                                         VCAL 330
   10 CONTINUE                                                          VCAL 340
      IIT=0                                                             VCAL 350
   20 CONTINUE                                                          VCAL 360
      VW(1)=VWALL                                                       VCAL 370
      DO 80 J=2,IE                                                      VCAL 380
      IF (IIT.GT.0) GO TO 70                                            VCAL 390
      IF (K.GT.1) GO TO 30                                              VCAL 400
      DGDW1=GW(J)                                                       VCAL 410
      DGDW2=GW(J-1)                                                     VCAL 420
      GO TO 50                                                          VCAL 430
   30 IF (LC.GT.1) GO TO 40                                             VCAL 440
```

DATE 05/12/77

```
      DGDW1=(G(2,J,2)-G(2,J,1))/DW                                      VCAL 450
      DGDW2=(G(2,J-1,2)-G(2,J-1,1))/DW                                  VCAL 460
      GO TO 50                                                          VCAL 470
40    DGDW1=(G(2,J,2)-G(2,J,1)+G(1,J,3)-G(1,J,2))/2.0D0/DW              VCAL 480
      DGDW2=(G(2,J-1,2)-G(2,J-1,1)+G(1,J-1,3)-G(1,J-1,2))/2.0D0/DW      VCAL 490
50    DFDXI1=0.0D0                                                      VCAL 500
      DFDXI2=0.0D0                                                      VCAL 510
      IF (LC.EQ.1) GO TO 60                                             VCAL 520
      DFDXI1=(F(2,J,2)-F(1,J,2))/DXI                                    VCAL 530
      DFDXI2=(F(2,J-1,2)-F(1,J-1,2))/DXI                                VCAL 540
      IF (LC.EQ.1) GO TO 60                                             VCAL 550
      W=UEW                                                             VCAL 560
      IF (LC.EQ.2) W=0.D0                                               VCAL 570
      DWDW=DUEWDW                                                       VCAL 580
      IF (LC.EQ.2) DWDW=DVEWDW                                          VCAL 590
      A=DSQRT(XI/2.D0/UEW)                                              VCAL 600
      A11=A*UEW*HWX(J)/HXOHXO(J)/HX(J)/HW(J)+A*UEW*HXX(1)/HX(J)**2-A*UEWVCAL 610
     1*HXX(J)/HXOHXO(J)/HX(J)**2+A*DUEWDX/2.D0/HXOHXO(J)/HX(J)+UEW/HXOHXVCAL 620
     2O(J)/HX(J)/2.D0/DSQRT(2.D0*UEW*XI)                                VCAL 630
      A12=A*UEW*HWX(J-1)/HXOHXO(J-1)/HX(J-1)/HW(J-1)+A*UEW*HXX(1)/HX(J-1VCAL 640
     1)**2-A*UEW*HXX(J-1)/HXOHXO(J-1)/HX(J-1)**2+A*DUEWDX/2.D0/HXOHXO(J-VCAL 650
     21)/HX(J-1)+UEW/HXOHXO(J-1)/HX(J-1)/2.D0/DSQRT(2.D0*UEW*XI)        VCAL 660
      A21=A*W*HXW(J)/HWOHWO(J)/HW(J)/HX(J)+A*W*HWW(1)/HW(J)**2-A*W*HWW(JVCAL 670
     1)/HWOHWO(J)/HW(J)**2-W*DSQRT(XI/2.D0/UEW**3)/2.D0/HWOHWO(J)*DUEWDWVCAL 680
     2/HW(J)+A*DWDW/HWOHWO(J)/HW(J)                                     VCAL 690
      A22=A*W*HXW(J-1)/HWOHWO(J-1)/HW(J-1)/HX(J-1)+A*W*HWW(1)/HW(J-1)**2VCAL 700
     1-A*W*HWW(J-1)/HWOHWO(J-1)/HW(J-1)**2-W*DSQRT(XI/2.D0/UEW**3)*DUEWDVCAL 710
     2W/2.D0/HWOHWO(J-1)/HW(J-1)+A*DWDW/HWOHWO(J-1)/HW(J-1)             VCAL 720
      A11=-A*A11                                                        VCAL 730
      A12=-A*A12                                                        VCAL 740
      A21=-A*A21                                                        VCAL 750
      A22=-A*A22                                                        VCAL 760
      A31=-XI/2.D0/HX(J)/HXOHXO(J)                                      VCAL 770
      A32=-XI/2.D0/HX(J-1)/HXOHXO(J-1)                                  VCAL 780
      A41=-XI*AL2/2.D0/HW(J)/HWOHWO(J)                                  VCAL 790
      A42=-XI*AL2/2.D0/HW(J-1)/HWOHWO(J-1)                              VCAL 800
      A51=A*BHV(J)/HX(J)/HW(J)                                          VCAL 810
      A52=A*BHV(J-1)/HX(J-1)/HW(J-1)                                    VCAL 820
      GO TO 70                                                          VCAL 830
60    CONTINUE                                                          VCAL 840
      A11=-1.D0/HXOHXO(J)/HX(J)                                         VCAL 850
      A12=-1.D0/HXOHXO(J-1)/HX(J-1)                                     VCAL 860
      A21=-AL1*B4/2.D0+AL1**2/2.D0-AL1*(HXW(J)/HX(J)+DHWOXW/HWX(1))/HWX(VCAL 870
     11)                                                                VCAL 880
      A22=-AL1*B4/2.D0+AL1**2/2.D0-AL1*(HWX(J-1)/HX(J-1)+DHWOXW/HWX(1))/VCAL 890
     1HWX(1)                                                            VCAL 900
      A31=0.D0                                                          VCAL 910
      A32=0.D0                                                          VCAL 920
      A41=-AL1/2.D0/HWX(1)                                              VCAL 930
      A42=A41                                                           VCAL 940
      A51=BHV(J)                                                        VCAL 950
```

```
DATE  05/12/77

      A52=BHV(J-1)
70    CONTINUE
      VW(J)=VW(J-1)+(A31*DFDXI1+A32*DFDXI2+A41*DGDW1+A42*DGDW2+A51*VW(J)
     1+A52*VW(J-1)+A11*FW(J)+A12*FW(J-1)+A21*GW(J)+A22*GW(J-1))*DETA(J)/
     22.D0
80    CONTINUE
      DIFV=0.D0
      DO 90 J=2,IE
      DIFVP=DABS((VW(J)-V(J))/VW(J))
      IF (DIFVP.GT.DIFV) DIFV=DIFVP
      V(J)=VW(J)
90    CONTINUE
      IF (DIFV.LE.CONV) GO TO 100
      IIT=1
      GO TO 20
100   CONTINUE
      RETURN
      END


      SUBROUTINE WALL
      IMPLICIT REAL*8(A-H,O-Z)
      REAL*8 NOSE
      COMMON /DEPVAR/ F(2,101,3),FN(2,101,3),G(2,101,3),GN(2,101,3),T(2,
     1101,3),TN(2,101,3),Y(101),YDL(101)
      COMMON /EDGE/ UEDG,TEDG,VEDG,DUEGDX,DUEGDW,D2UEDW,DVEGDX,DVEGDW,RE
     1DG,DRDGDX,ZBOD,XEDG
      COMMON /EDGW/ UEW,VEW,TEW,DUEWDX,DUEWDW,D2UEWW,DVEWDX,DVEWDW,REW
      COMMON /FRSTRM/ RHOINF,PINF,TFS,UFS,AMUINF,CP,REINF,PR
      COMMON /GEOM/ ALPHA,NOSE,RNOSE,WLST,X,XX,WX
      COMMON /IECOEF/ B1,B2,B3,B4,B5,B6,AL1,AL2,AL3,XLAMB
      COMMON /INJECT/ INJCT,NOINJ,MASTRN
      COMMON /INTEGR/ IE,IM,KEND,KEND2,KLX,K,L,NBLNT1,IND,KPRT,LPRT,KPR,
     1LPR
      COMMON /SOLPNT/ VW(101),GW(101),TW(101),GWN(101),FWN(101),FW(101),
     1TWN(101),XIM,XW,RW
      COMMON /STAG/ PSTAG,TSTAG,PNC,QWSTAG
      COMMON /SURFAS/ CWALL,CWIND,PEWIND,VWALL,TWALL,XTH(500),TWX(500),X
     1CI(500),CIX(500),TCONW,KCI,KTH
      COMMON /TMPRTR/ TEMP(101),TP(101),RTW,TB
      COMMON /XICORD/ XI,XXI,DXI,XIOLD
      DATA BLUNT,SHARP/5HBLUNT,5HSHARP/
      DATA ABL,TRA/3HABL,3HTRA/
      PI=DARCOS(-1.0D0)
C
C     INTERPOLATE FOR VALUES OF CWALL AT THE WINDWARD STREAMLINE
C
      EPSVD=1.D0/DSQRT(REINF)
```

DATE 05/12/77

```
      IF (MASTRN.E).0) GO TO 30
      IF (K.GT.1) GO TO 20
      PEWIND=PEW
      IF (KCI.EQ.0) GO TO 20
      IF (XW.GT.XCI(KCI)) GO TO 30
      J=0
10    J=J+1
      IF (XW.GT.XCI(J)) GO TO 10
      IF (J.LT.2) J=2
      IF (J.GT.KCI-1) J=KCI-1
      CALL INTER3 (XW,XCI(J-1),XCI(J),XCI(J+1),CIX(J-1),CIX(J),CIX(J+1),
     1CWIND)
      CWALL=CWIND
      GO TO 40
20    CONTINUE
      IF (WX.LT.90.0D0) GO TO 40
30    CWALL=0.0D0
40    CONTINUE
C
C     INTERPOLATE FOR VALUES OF THALL AT THE WINDWARD STREAMLINE
C
      IF (K.GT.1) GO TO 70
      IF (KTW.EQ.0) GO TO 60
      J=0
50    J=J+1
      IF (XW.GT.XTW(J)) GO TO 50
      IF (J.LT.2) J=2
      IF (J.GT.KTW-1) J=KTW-1
      CALL INTER3 (XW,XTW(J-1),XTW(J),XTW(J+1),TWX(J-1),TWX(J),TWX(J+1),
     1TWALL)
      GO TO 70
60    TWALL=RTW
70    CONTINUE
C
C     CALCULATE THE VALUE OF BIG V AT THE WALL
C
      VWALL=0.D0
      IF (NOSE.EQ.SHARP.AND.XIW.EQ.0.0D0) GO TO 90
      IF (XIW.EQ.0.0D0) GO TO 80
      VWALL=DSQRT(XI/2.D0/UEW)*CWALL
      GO TO 90
80    VWALL=DSQRT(.5D0/DUEWDX)*CWALL
90    CONTINUE
      RETURN
      END


      SUBROUTINE XMOM (LC)
```

DATE 05/12/77

```
      IMPLICIT REAL*8(A-H,O-Z)                                          XMOM  20
      REAL*8 NOSE                                                       XMOM  30
      COMMON /DEPVAR/ F(2,101,3),FN(2,101,3),G(2,101,3),GN(2,101,3),T(2,XMOM  40
     1101,3),TN(2,101,3),Y(101),YOL(101)                                XMOM  50
      COMMON /GEOM/ ALPHA,NOSE,RNOSE,WLST,X,XX,WX                       XMOM  60
      COMMON /IECOEF/ B1,B2,B3,B4,B5,B6,AL1,AL2,AL3,XLAMB               XMOM  70
      COMMON /INTEGR/ IE,IM,KEND,KEND2,KLX,K,L,NBLNT1,IND,KPRT,LPRT,KPR,XMOM  80
     1LPR                                                               XMOM  90
      COMMON /PDECOF/ A0(101),A1(101),A2(101),A3(101),A4(101),A5(101)   XMOM 100
      COMMON /SOLPNT/ VW(101),GW(101),TW(101),GWN(101),FWN(101),FW(101),XMOM 110
     1TWN(101),XIW,XW,RW                                                XMOM 120
      COMMON /TRANSN/ KTRANS,KONSET,XIF,CHI2(101),CHIMAX,XBAR           XMOM 130
      COMMON /TRBLNT/ ASTAR,AKSTAR,ALAMDA,YSUBL,EVSCTY(101),EDYLAW,EPLUSXMOM 140
     1(101),LAMTRB                                                      XMOM 150
      COMMON /XICORD/ XI,XXI,DXI,XIOLD                                  XMOM 160
      COMMON /ZCOORD/ ETAINF,ETAFAC,ETA(101),DETA(101),ADTEST,KADETA    XMOM 170
      COMMON /NCOORD/ XN(101),ETAX,ETAW                                 XMOM 180
      COMMON /INTGRL/ XIFNX(101),XIFNW(101),XIFNXX(101),XIFNWW(101)     XMOM 190
      COMMON /EDGW/ UEW,VEW,TEW,DUEWDX,DUEWDW,D2UEWW,DVEWDX,DVEWDW,REW  XMOM 200
      COMMON /GECOEF/ HXOHXO(101),HWOHWO(101),BHV(101),BHM(101),BHT(101)XMOM 210
     1,XKX(101),XKW(101),BMX(101),BMW(101),XLX(101),XLW(101),BAFF(101),BXMOM 220
     2AGF(101),BAFG(101),BAGG(101),XKXOAL(101)                          XMOM 230
      COMMON /HETA/ HX(101),HW(101),HXX(101),HXW(101),HXWW(101),HXN(101)XMOM 240
     1,HXNN(101),HWX(101),HWW(101),HWN(101),HWNN(101)                   XMOM 250
      DIMENSION ROMU1(101), ROMU1N(101)                                 XMOM 260
      DATA SHARP,BLUNT/5HSHARP,5HBLUNT/                                 XMOM 270
C                                                                       XMOM 280
C     SUBROUTINE SETS UP THE COEFFICIENTS OF THE PARTIAL DIFFERENTIAL   XMOM 290
C     X MOMENTUM EQUATION                                               XMOM 300
C                                                                       XMOM 310
      DO 10 J=1,IE                                                      XMOM 320
      ROMU1(J)=1.0D0+XIF*EPLUS(J)                                       XMOM 330
   10 CONTINUE                                                          XMOM 340
      CALL DERIV (ROMU1,ETA,IE,1,ROMU1N)                                XMOM 350
      CALL PRESSI (1)                                                   XMOM 360
      DO 30 J=1,IE                                                      XMOM 370
      A0(J)=ROMU1(J)                                                    XMOM 380
      A1(J)= ROMU1N(J)-BHM(J)*ROMU1(J)-VW(J)                            XMOM 390
      A2(J)=-BMX(J)*RCMU1(J)+ROMU1N(J)*HXN(J)/HX(J)                     XMOM 400
      A4(J)=-XI*HX(1)*FW(J)/2.D0/HX(J)**2                               XMOM 410
      A5(J)=0.D0                                                        XMOM 420
      IF (LC.EQ.1) GO TO 20                                             XMOM 430
      A2S=(XLX(1)-XLX(J))/2.D0-BAFF(J)+.5D0*B1/HXOHXO(J)**2             XMOM 440
      A2(J)=A2(J)-2.D0*A2S*FW(J)                                        XMOM 450
      A3(J)=.5D0*B1/HXOHXO(J)**2+(XLX(1)-XLX(J))/2.D0-BAFF(J)+XI/2.D0/HXXMOM 460
     1(1)/UEW**2*XIFNX(J)+A2S*FW(J)**2                                  XMOM 470
      IF (LC.EQ.2) GO TO 30                                             XMOM 480
      A2(J)=A2(J)-AL2*GW(J)/2.D0/HWOHWO(J)**2*(B2+XKX(1))               XMOM 490
      A3(J)=A3(J)+AL1*B2/HWOHWO(J)**2/2.D0-XKW(J)*AL2**2*HXOHXO(J)*(AL3*XMOM 500
     1*2-GW(J)**2)/2.D0/HWOHWO(J)**2+XKX(1)*AL1/2.D0/HWOHWO(J)**2-BAGF(JXMOM 510
     2)*(AL3**2-GW(J)**2)                                               XMOM 520
```

```
DATE  05/12/77

      A5(J)=-XI*GW(J)*AL2/2.DO/HW(J)/HWOHWO(J)                          XMOM 530
      GO TO 30                                                          XMOM 540
20    CONTINUE                                                          XMOM 550
      A2(J)=A2(J)-.5DO*FW(J)/HXOHXO(J)**2-AL1**2*GW(J)                  XMOM 560
      A3(J)=.5DO/HXOHXO(J)**2+AL2**2*HXOHXO(J)*(1.DO-GW(J)**2)/2.DO/HX(JXMOM 570
     1)+AL1**2/2.DO+DSQRT(2.DO)*(XIFNXX(J)+DVEWDX*XIFNWW(J)/DSQRT(DUEWDXXMOM 580
     2))/DUEWDX**2                                                      XMOM 590
      A4(J)=0.DO                                                        XMOM 600
      A5(J)=-AL1*GW(J)/HWX(1)                                           XMOM 610
30    CONTINUE                                                          XMOM 620
      IRITE=0                                                           XMOM 630
      IF (IRITE.EQ.0) GO TO 50                                          XMOM 640
      WRITE (6,60)                                                      XMOM 650
      DO 40 I=1,IE,5                                                    XMOM 660
      WRITE (6,70) A0(I),A1(I),A2(I),A3(I),A4(I),A5(I)                  XMOM 670
40    CONTINUE                                                          XMOM 680
50    CONTINUE                                                          XMOM 690
      RETURN                                                            XMOM 700
C                                                                       XMOM 710
60    FORMAT (1H0,12HIN XMOM A0-5)                                      XMOM 720
70    FORMAT (1H0,6E13.6)                                               XMOM 730
      END                                                               XMOM 740
```